Proceedings

THIRTY-SEVENTH ANNUAL ALLERTON CONFERENCE ON COMMUNICATION, CONTROL AND COMPUTING

September 22 - 24, 1999

Allerton House, Monticello, Illinois
Sponsored by the
Coordinated Science Laboratory and the
Department of Electrical and Computer Engineering of the
University of Illinois at Urbana-Champaign

UNIVERSITY OF ILLINOIS
LIBRARY
AT URBANA-CHAMPAIGN
ENGINEERING

PROCEEDINGS

THIRTY-SEVENTH ANNUAL ALLERTON CONFERENCE ON COMMUNICATION, CONTROL, AND COMPUTING

Bruce Hajek
R.S. Sreenivas
Conference Co-Chairs

Conference held
September 22, September 23, and September 24, 1999
Allerton House
Monticello, Illinois

Sponsored by
The Coordinated Science Laboratory
and
The Department of Electrical and Computer Engineering
of the
UNIVERSITY OF ILLINOIS
at
Urbana-Champaign

Digitized by the Internet Archive
in 2026 with funding from
University of Illinois Urbana-Champaign

https://archive.org/details/proceedings1999alle_0

621.3811
A25
1999

ENX

FOREWORD

These Proceedings contain papers that were presented at the Thirty-seventh Annual Allerton Conference on Communication, Control, and Computing which was held at Allerton House, Monticello, Illinois, on September 22, thru September 24, 1999. As in past years, the Conference was sponsored by the Coordinated Science Laboratory and the Department of Electrical and Computer Engineering of the University of Illinois at Urbana-Champaign.

104 invited papers and 61 contributed papers were presented at the Conference in twenty-eight sessions. We thank all the authors and particularly the speakers for contributing to the success of the Conference. We are also grateful to our colleagues at the University of Illinois at Urbana-Champaign: Professors T. Başar, R.E. Blahut, C. Beck, N. Boston, G. Dullerud, R. Koetter, U. Madhow, M. Médard, S.P. Meyn, D. Sarwate, A. Singer, M.W. Spong, and R. Srikant, for organizing many of the invited sessions, reviewing the contributed papers, chairing sessions, and helping in many other ways in the smooth running of the Conference. Additional invited sessions were organized by Professor Y. Shavitt, Bell Labs, Lucent Technologies; M. Feder, Tel Aviv University; E. Modiano, Massachusetts Institute of Technology; V.V. Veeravalli, Cornell University; A. Nehorai, University of Illinois at Chicago; and S. Reveliotis, Georgia Institute of Technology; we are especially grateful to them for their efforts on behalf of the Conference. Professors S. Lin, University of Hawaii; R. Cruz, University of California, San Diego; and M.K. Varanasi, University Colorado at Boulder, chaired contributed sessions at the Conference, and for this we thank them.

The plenary speaker at the Conference was Frank Kelly of Cambridge University, England, who gave an illuminating lecture titled "Proportional Fairness, Resource Pricing and the Evolution of Congestion Control." We are grateful to him for his excellent presentation. We are also indebted to the entertainment from the "Pocket Big Band," for giving us a wonderful evening of classical music.

We thank Mrs. Beverly Becker and her staff at Allerton House for cheerfully accommodating our Conference and for their constant attention to our needs. It is also a great pleasure to thank all the staff members of the Coordinated Science Laboratory who so ably handled all the practical aspects of organizing the Conference. In particular, we wish to mention Mr. Steven Williams and Ms. Susan Baxley who handled all the business affairs, and Mr. Dan Jordan who handled the sales and shipping of these Proceedings. Most of all, however, we would like to thank our Conference Secretary, Mrs. Becky Lonberger, who with the able assistance of Mrs. Francie Bridges and Mrs. Terri Hovde worked all year round in handling announcements, calls for papers, program preparation, Allerton House arrangements, registration, and the final editing of these Proceedings. We would also like to thank Mrs. Nancy Carr for her help at the registration desk. The lion's share of the credit for the success of the Conference belongs to these individuals and it gives us the greatest pleasure to acknowledge this here.

We close this foreword with the traditional countdown for the reopening of the time capsule buried Tuesday, September 30, 1997, on the occasion of the Thirty-Fifth Annual Allerton Conference. The number of years to go is 33.

Bruce Hajek and R.S. Sreenivas

Contin.

Table 1: 35 YEARS OF ALLERTON CONFERENCES

Name	Dates	Chair(s)
First Annual Allerton Conference on Circuit and System Theory	November 15-17, 1963	J. B. Cruz, Jr.
Second Annual Allerton Conference on Circuit and System Theory	September 28-30, 1964	W. R. Perkins
Third Annual Allerton Conference on Circuit and System Theory	October 20-22, 1965	M. E. VanValkenburg
Fourth Annual Allerton Conference on Circuit and System Theory	October 5-7, 1966	William R. Perkins J. B. Cruz, Jr.
Fifth Annual Allerton Conference on Circuit and System Theory	October 4-6, 1967	J. B. Cruz, Jr. T. N. Trick
Sixth Annual Allerton Conference on Circuit and System Theory	October 2-4, 1968	T. N. Trick R. T. Chien
Seventh Annual Allerton Conference on Circuit and System Theory	October 8-10, 1969	A. H. Haddad G. Metze T. N. Trick
Eighth Annual Allerton Conference on Circuit and System Theory	October 7-9, 1970	A. H. Haddad B. C. Kuo G. Metze
Ninth Annual Allerton Conference on Circuit and System Theory	October 6-8, 1971	J. B. Cruz, Jr. A. H. Haddad
Tenth Annual Allerton Conference on Circuit and System Theory	October 4-6, 1972	J. B. Cruz, Jr. C. L. Chen
Eleventh Annual Allerton Conference on Circuit and System Theory	October 3-5, 1973	W. R. Perkins P. V. Kokotovic
Twelfth Annual Allerton Conference on Circuit and System Theory	October 2-4, 1974	P. V. Kokotovic E. S. Davidson
Thirteenth Annual Allerton Conference on Circuit and System Theory	October 1-3, 1975	E. S. Davidson T. N. Trick
Fourteenth Annual Allerton Conference on Circuit and System Theory	September 29-October 1, 1976	T. N. Trick M. E. VanValkenburg
Fifteenth Annual Allerton Conference on Communication, Control, and Computing	September 28-30, 1977	F. P. Preparata M. B. Pursley
Sixteenth Annual Allerton Conference on Communication, Control, and Computing	October 4-6, 1978	M. B. Pursley J. B. Cruz, Jr.
Seventeenth Annual Allerton Conference on Communication, Control, and Computing	October 10-12, 1979	J. B. Cruz, Jr. F. P. Preparata
Eighteenth Annual Allerton Conference on Communication, Control, and Computing	October 8-10, 1980	D. V. Sarwate W. R. Perkins

Nineteenth Annual Allerton Conference on Communication, Control, and Computing	September 30-October 2, 1981	W. K. Jenkins D. V. Sarwate
Twentieth Annual Allerton Conference on Communication, Control, and Computing	October 6-8, 1982	H. V. Poor W. K. Jenkins
Twenty-First Annual Allerton Conference on Communication, Control, and Computing	October 5-7, 1983	Tamer Başar Donna J. Brown H. Vincent Poor
Twenty-Second Annual Allerton Conference on Communication, Control, and Computing	October 3-5, 1984	David C. Munson, Jr. Tamer Başar
Twenty-Third Annual Allerton Conference on Communication, Control, and Computing	October 2-4, 1985	Bruce Hajek David C. Munson, Jr.
Twenty-Fourth Annual Allerton Conference on Communication, Control, and Computing	October 1-3, 1986	Michael C. Loui Juraj V. Medanic
Twenty-Fifth Annual Allerton Conference on Communication, Control, and Computing	September 30-October 2, 1987	P. R. Kumar Michael C. Loui
Twenty-Sixth Annual Allerton Conference on Communication, Control, and Computing	September 28-30, 1988	Mark W. Spong Bruce Hajek
Twenty-Seventh Annual Allerton Conference on Communication, Control, and Computing	September 27-29, 1989	Juraj V. Medanic P. R. Kumar
Twenty-Eighth Annual Allerton Conference on Communication, Control, and Computing	October 3-5, 1990	Donna J. Brown Kameshwar R. Poolla
Twenty-Ninth Annual Allerton Conference on Communication, Control, and Computing	October 2-4, 1991	George Cybenko Constantine Polychronopoulos
Thirtieth Annual Allerton Conference on Communication, Control, and Computing	September 30-October 2, 1992	Paul Van Dooren Mark W. Spong
Thirty-First Annual Allerton Conference on Communication, Control, and Computing	September 29-October 1, 1993	Dilip V. Sarwate Paul Van Dooren
Thirty-Second Annual Allerton Conference on Communication, Control, and Computing	September 28-30, 1994	W. Kenneth Jenkins Dilip V. Sarwate
Thirty-Third Annual Allerton Conference on Communication, Control, and Computing	October 4-6, 1995	Sean Meyn Ken Zeger
Thirty-Fourth Annual Allerton Conference on Communication, Control, and Computing	October 2-4, 1996	Sean P. Meyn W. Kenneth Jenkins
Thirty-Fifth Annual Allerton Conference on Communication, Control, and Computing	September 29-October 1, 1997	Tamer Başar Alexander Vardy
Thirty-Sixth Annual Allerton Conference on Communication, Control, and Computing	September 23-25, 1998	Tamer Başar Bruce Hajek
Thirty-Seventh Annual Allerton Conference on Communication, Control, and Computing	September 22-24, 1999	Bruce Hajek R.S. Sreenivas

TABLE OF CONTENTS

FORWORD .... *i*

I-A: STOCHASTIC NETWORKS I

Organizers: S.P. Meyn and R. Srikant
(University of Illinois at Urbana-Champaign)
Chair: S.P. Meyn
(University of Illinois at Urbana-Champaign)

REPRESENTATION AND EXPANSION OF (MAX, PLUS) LYAPUNOV EXPONENTS 1
F. Baccelli, S. Gaubert, and D. Hong

MARTINGALE PROBLEMS AND LINEAR PROGRAMS FOR SINGULAR CONTROL 11
T.G. Kurtz and R.H. Stockbridge

STATIONARY REFLECTED LÉVY PROCESSES IN STOCHASTIC NETWORKS 21
T. Konstantopoulos and G. Last

ON THE IMPACT OF VARIABILITY ON THE BUFFER DYNAMICS IN IP NETWORKS 30
Y. Joo, V. Ribeiro, A. Feldmann, A.C. Gilbert, and W. Willinger

QUEUEING NETWORKS WITH INTERACTING SERVICE RESOURCES 42
M. Armony and N. Bambos

I-B: CODING THEORY I: DECODING AND CHANNELS

Organizers: R. Koetter and R.E. Blahut
(University of Illinois at Urbana-Champaign)
Chair: A. Vardy
(University of California, San Diego)

A NEW UPPER BOUND ON THE RELIABILITY FUNCTION OF THE GAUSSIAN CHANNEL 52
A. Ashikhmin, A. Barg, and S. Litsyn

RECURSIVE DECODING OF REED-MULLER CODES 61
I. Dumer

LOSSLESS COMPRESSION IN CONSTRAINED CODING 70
J.L. Fan, B. Marcus, and R. Roth

I-C: HYBRID/DISCRETE-EVENT-DYNAMIC SYSTEMS

Chair: R.S. Sreenivas
(University of Illinois at Urbana-Champaign)

MODELLING OF TIMED DISCRETE EVENT SYSTEMS 75
R.S. Minhas and W.M. Wonham

INTERACTING DISCRETE EVENT SYSTEMS 85
S. Abdelwahed and W.M. Wonham

STABILITY ANALYSIS FOR INTERCONNECTED HYBRID SYSTEMS 93
S. Yamamoto and T. Ushio

DECENTRALIZED SUPERVISORY CONTROL OF CONCURRENT DISCRETE EVENT SYSTEMS WITH PARTIAL OBSERVATIONS 103
S. Jiang and R. Kumar

A NEW PROBABILISTIC APPROACH TO CONGESTION CONTROL IN COMMUNICATION NETWORKS 113
H. Mortazavian and J. Mirkovic

A BIGRAPH MATCHING THEOREM 124
S. Ayyorgun and R.L. Cruz

I-D: ACTIVE NETWORKS

Organizer/Chair: Y. Shavitt
(Bell Labs, Lucent Technologies)

CHUNKS IN PLAN: LANGUAGE SUPPORT FOR PROGRAMS AS PACKETS 127
J.T. Moore, M. Hicks, and S. Nettles

ON THE INTERFACE OF PROGRAMMABLE NETWORK ELEMENTS 137
G. Hjálmtýsson

BOWMAN AND CANES: IMPLEMENTATION OF AN ACTIVE NETWORK 147
S. Merugu, S. Bhattacharjee, Y. Chae, M. Sanders, K. Calvert, and E. Zegura

DESIGN OF A FLEXIBLE OPEN PLATFORM FOR HIGH PERFORMANCE ACTIVE NETWORKS 157
S. Choi, D. Decasper, J. Dehart, R. Keller, J. Lockwood, J. Turner, and T. Wolf

IMPLEMENTING A CONCAST SERVICE 166
K. Calvert, J. Griffioen, B. Mullins, A. Sehgal, and S. Wen

ACTIVE DISTRIBUTED MANAGEMENT FOR IP NETWORKS 176
R. Kawamura and R. Stadler

I-F: SPACE-TIME METHODS FOR COMMUNICATION

Chair: D. Sarwate
(University of Illinois at Urbana-Champaign)

NEW APPROACH FOR SPACE-TIME TRANSMITTER/RECEIVER DESIGN 186
H. El Gamal and A.R. Hammons, Jr.

INTERFERENCE SUPPRESSION FOR CDMA VIA A SPACE-TIME POWER MINIMIZATION BASED PREPROCESSOR WITH APPLICATIONS TO GPS 196
W.L. Myrick, M.D. Zoltowski, and J.S. Goldstein

SOFT-WEIGHTED TRANSMIT DIVERSITY FOR WCDMA 204
A. Hottinen, R. Wichman, and D. Rajan

MULTIUSER DETECTION TECHNIQUES FOR COMBINED ARRAY PROCESSING AND SPACE-TIME BLOCK CODING 214
B. Lu and X. Wang

A TRANSMIT ADAPTIVE ANTENNA SCHEME WITH FEEDBACK FOR WIRELESS COMMUNICATIONS 216
Y. Le Pézennec, F. Boixadera, Y. Farmine, and N. Whinnett

II-A: CODING THEORY II: ITERATIVE DECODING AND TURBO CODES

Organizers: R. Koetter and R.E. Blahut
(University of Illinois at Urbana-Champaign)
Chair: R. Koetter
(University of Illinois at Urbana-Champaign)

CONCENTRATE 221
T. Richardson and R. Urbanke

EFFICIENT ENCODING OF LOW-DENSITY PARITY-CHECK CODES 231
T. Richardson and R. Urbanke

IRREGULAR TURBOCODES 241
B.J. Frey and D.J.C. MacKay

ON QUASI-CYCLIC REPEAT-ACCUMULATE CODES 249
R.M. Tanner

THE SERIAL CONCATENATION OF RATE-1 CODES THROUGH UNIFORM RANDOM INTERLEAVERS 260
H.D. Pfister and P.H. Siegel

II-B: STOCHASTIC NETWORKS II

Organizers: S.P. Meyn and R. Srikant
(University of Illinois at Urbana-Champaign)
Chair: R. Srikant
(University of Illinois at Urbana-Champaign)

QUEUE LENGTH ASYMPTOTICS FOR MARKOVIAN SERVICE NETWORKS 270
A. Mandelbaum, W.A. Massey, and M.I. Reiman

EXACT ASYMPTOTICS FOR 1-LIMITED EXPONENTIAL POLLING MODELS 280
W. Chang, D.G. Down, and R.D. Foley

INVARIANT RATE FUNCTIONS FOR DISCRETE TIME QUEUES 288
A.J. Ganesh, N. O'Connell, and B. Prabhakar

LARGE DEVIATIONS AND OPTIMALITY OF THE LARGEST WEIGHTED DELAY FIRST DISCIPLINE 297
A.L. Stolyar and K. Ramanan

ON ESTIMATING BUFFER OVERFLOW PROBABILITIES UNDER MARKOV-MODULATED INPUTS 306
I.Ch. Paschalidis and S. Vassilaras

INDUCED BURSTINESS IN GENERALIZED PROCESSOR SHARING QUEUES WITH LONG-TAILED TRAFFIC FLOWS 316
S. Borst, O. Boxma, and P. Jelenković

THE ASYMPTOTICS OF SELECTING THE SHORTEST OF TWO, IMPROVED 326
M. Mitzenmacher and B. Vöcking

II-C: LEARNING ALGORITHMS IN SIGNAL PROCESSING

Organizers: A. Singer and M. Feder
(University of Illinois at Urbana-Champaign and Tel Aviv University)
Chair: M. Feder
(Tel Aviv University)

UNIVERSAL FILTERING AND PREDICTION OF INDIVIDUAL SEQUENCES CORRUPTED BY NOISE 328
A. Baruch and N. Merhav

FAST RLS LAGUERRE ADAPTIVE FILTERING 338
R. Merched and A.H. Sayed

MACHINE LEARNING APPLICATIONS IN GRID COMPUTING 348
G. Cybenko, G. Jiang, and D. Bilar

REDUNDANCY OF THE LEMPEL-ZIV CODES 358
S.A. Savari

THE INFORMATION BOTTLENECK METHOD 368
N. Tishby, F.C. Pereira, and W. Bialek

THEORY MEETS PRACTICE: UNIVERSAL SOURCE CODING WITH THE BURROWS WHEELER TRANSFORM 378
M. Effros

II-D: OPTICAL NETWORKS I

Organizers: M. Médard and E. Modiano
(University of Illinois at Urbana-Champaign
and Massachusetts Institute of Technology)
Chair: E. Modiano
(Massachusetts Institute of Technology)

OPTICAL SPACE COMMUNICATIONS AND NETWORKING 388
V.W.S. Chan

ON THE BENEFITS OF CONFIGURABILITY IN WDM NETWORKS 390
E. Modiano and A. Narula-Tam

NONBLOCKING WDM NETWORKS WITH FIXED-TUNED TRANSMITTERS AND TUNABLE RECEIVERS 400
T. Lin and G. Sasaki

ON NEW ARCHITECTURES FOR WDM NETWORKS 402
A. Sen, T. Shah, and B.P. Sinha

ALL-OPTICAL LABEL SWAPPING WITH WAVELENGTH CONVERSION FOR WDM-IP NETWORKS WITH SUBCARRIER MULTIPLEXED ADDRESSING 414
D.J. Blumenthal

II-E: COMMUNICATION SYSTEMS AND SERVICES

Chair: S. Lin
(University of Hawaii)

ON DISCRETE SUFFICIENT STATISTICS FOR ACQUISITION IN ASYNCHRONOUS BAND-LIMITED CDMA SYSTEMS 424
A. Mantravadi and V.V. Veeravalli

FREQUENCY SYNCHRONIZATION ALGORITHM FOR FREQUENCY HOPPING SYSTEM BASED ON SINGULAR VALUE DECOMPOSITION 434
A. Pouttu

A SOFTWARE-ORIENTED STREAM CIPHER FOR CELLULAR AND PERSONAL COMMUNICATIONS SERVICES 443
M. Zhang, A. Chan, and C. Carroll

BINARY RANK CRITERIA FOR PSK MODULATED SPACE-TIME CODES 445
H. El Gamal and A.R. Hammons, Jr.

TURBO CODES WITH ORTHOGONAL MODULATION IN DS-CDMA MOBILE RADIO SYSTEM WITH SHORT FRAME TRANSMISSION 451
G. Li and Y.L. Guan

AN INTERACTIVE CONCATENATED TURBO CODING SYSTEM 461
Y. Liu, H. Tang, S. Lin, and M.P.C. Fossorier

BI-DIRECTIONAL SOVA DECODING FOR TURBO-CODES 471
J. Chen, M.P.C. Fossorier, S. Lin, and C. Xu

II-F: FADING CHANNELS AND POWER CONTROL

Chair: D. Sarwate
(University of Illinois at Urbana-Champaign)

ANALYSIS OF AN UP/DOWN POWER CONTROL ALGORITHM IN CDMA REVERSE LINK UNDER FADING CONDITION 475
L. Song, N. Mandayam, and Z. Gajic

A CLASS OF DISTRIBUTED ASYNCHRONOUS POWER CONTROL ALGORITHMS FOR CELLULAR WIRELESS SYSTEMS 485
J.D. Herdtner and E.K.P. Chong

DISTRIBUTED CONNECTION ADMISSION CONTROL FOR POWER-CONTROLLED CELLULAR WIRELESS SYSTEMS 495
M. Xiao, N.B. Shroff, and E.K.P. Chong

INTERFERENCE AVOIDANCE AND DISPERSIVE CHANNELS: A NEW LOOK AT MULTICARRIER MODULATION 505
D.C. Popescu and C. Rose

PERFORMANCE OF OPTIMAL CODES ON GAUSSIAN AND RAYLEIGH FADING CHANNELS: A GEOMETRICAL APPROACH 515
S. Vialle and J. Boutros

IMPROVED MARKOV MODELS FOR FADING CHANNELS: ANALYSIS AND DESIGN 525
D.L. Goeckel, M.J. Chu, and W.E. Stark

III-A: CODING THEORY III: ALGEBRAIC AND COMBINATORIAL CODING THEORY

Organizers: R. Koetter and R.E. Blahut
(University of Illinois at Urbana-Champaign)
Chair: N. Boston
(University of Illinois at Urbana-Champaign)

ON THE CLASSIFICATION OF EXTREMAL ADDITIVE CODES OVER GF(4) 535
P. Gaborit, W.C. Huffman, J.-L. Kim, and V. Pless

TWO FAST ALGORITHMS IN THE SUDAN DECODING PROCEDURE 545
G.-L. Feng

FROM WEIGHT ENUMERATORS TO ZETA FUNCTIONS 555
I. Duursma

ALTERNATIVE APPROACHES TO THE COMPUTATION OF ERROR VALUES FOR HERMITIAN CODES 557
M.E. O'Sullivan

III-B: STOCHASTIC NETWORKS III

Organizers: S.P. Meyn and R. Srikant
(University of Illinois at Urbana-Champaign)
Chair: S.P. Meyn
(University of Illinois at Urbana-Champaign)

BUFFER OVERFLOW ASYMPTOTICS IN HOL SERVICE SYSTEMS WITH HETEROGENEOUS LONG-TAILED INPUTS 567
C. Kotopoulos, N. Likhanov and R.R. Mazumdar

SCHEDULING AND CONTROL OF MANUFACTURING SYSTEMS — A FLUID APPROACH 577
G. Weiss

MULTICLASS NETWORKS IN HEAVY TRAFFIC: ASYMPTOTIC OPTIMALITY OF TRACKING POLICIES 587
C. Maglaras

SCHEDULING OPEN QUEUEING NETWORKS WITH SUFFICIENTLY FLEXIBLE RESOURCES 597
S. Kumar

OPTIMALLY STABILIZING CONTROLS FOR A DETERMINISTIC NETWORK MODEL 607
P. Dupuis and R. Atar

III-C: ROBUST CONTROL AND DECISION MAKING

Chair: C. Beck
(University of Illinois at Urbana-Champaign)

EVALUATING CUMULANT CONTROLLERS ON A BENCHMARK STRUCTURE PROTECTION PROBLEM IN THE PRESENCE OF CLASSIC EARTHQUAKES 617
K.D. Pham, M.K. Sain, S.R. Liberty, and B.F. Spencer, Jr.

RISK-SENSITIVE DECISION-THEORETIC TROUBLESHOOTING 627
M.A. Shayman and E. Fernández-Gaucherand

H_∞ CONTROL FOR MIXED DISTURBANCE REJECTION 637
J.C. Luo and E.B. Lee

SOLVING POLYNOMIAL SYSTEMS IN ROBUST STABILITY ANALYSIS 641
N.-P. Ke

ROBUST, NEAR TIME-OPTIMAL CONTROL OF THIRD-ORDER UNCERTAIN SYSTEMS 651
K.H. You and E.B. Lee

A NEW CONVEX RELAXATION FOR ROBUST H_2 PERFORMANCE ANALYSIS OF UNCERTAIN LINEAR SYSTEMS 655
L. El Ghaoui and E. Feron

A NEW RESULT ON THE BELLMAN EQUATION FOR EXIT TIME CONTROL PROBLEMS WITH CRITICAL GROWTH DYNAMICS 657
M. Malisoff

III-D: OPTICAL NETWORKS II

Organizers: M. Médard and E. Modiano
(University of Illinois at Urbana-Champaign and
Massachusetts Institute of Technology)
Chair: M. Médard
(University of Illinois at Urbana-Champaign)

FAULT PROTECTION IN WDM MESH NETWORKS 659
G. Ellinas

A COMPARISON OF ALLOCATION POLICIES IN WAVELENGTH ROUTING NETWORKS 669
Y. Zhu, G.N. Rouskas, and H.G. Perros

OPTICAL BUFFERS FOR MULTI-TERABIT IP ROUTERS 679
D.K. Hunter, I. Andonovic, and M.C. Chia

THE λ-SCHEDULER: A MULTIWAVELENGTH SCHEDULING SWITCH 689
J.P. Lang, E.A. Varvarigos, and D.J. Blumenthal

ON DIFFERENT ROUTING STRATEGIES IN TRANSPARENT ALL-OPTICAL NETWORKS 699
O.K. Tonguz

III-E: COMMUNICATION NETWORKS

Chair: R. Cruz
(University of California, San Diego)

TRANSMISSION POLICIES FOR TIME VARYING CHANNELS WITH AVERAGE DELAY CONSTRAINTS 709
B.E. Collins and R.L. Cruz

FAIR ALLOCATION OF UTILITIES IN MULTIRATE MULTICAST NETWORKS 718
S. Sarkar and L. Tassiulas

ON THE USE OF MULTIPLE WORKING POINTS IN MULTICHANNEL ALOHA WITH DEADLINES 728
D. Baron and Y. Birk

SPECIFICATION AND ANALYSIS OF A RELIABLE BROADCASTING PROTOCOL IN MAUDE 738
G. Denker, J.J. Garcia-Luna-Aceves, J. Meseguer, P.C. Ölveczky, J. Raju, B. Smith, and C.L. Talcott

MODELING AND ANALYSIS OF ACTIVE MESSAGES IN VOLATILE NETWORKS 748
C. Okino and G. Cybenko

IMPLEMENTATION OF AN ACTIVE CONGESTION CONTROL SCHEME IN NARROWBAND ATM NETWORKS 758
S. Sheth, J. Evans, A. Kulkarni, and G. Minden

CPU SCHEDULING FOR ACTIVE PROCESSING USING FEEDBACK DEFICIT ROUND ROBIN 768
T. Wolf and D. Decasper

III-F: WIRELESS COMMUNICATION I: DETECTION AND ESTIMATION

Organizers: V.V. Veeravalli and U. Madhow
(Cornell University and University of Illinois at Urbana-Champaign)
Chair: V.V. Veeravalli
(Cornell University)

PRECODING FOR SCATTERING FUNCTION ESTIMATION OF MOBILE CHANNELS USING OUTPUT CORRELATIONS ONLY 770
C. Tepedelenlioğlu and G.B. Giannakis

TWO-STAGE HYBRID ACQUISITION OF MULTICARRIER DIRECT-SEQUENCE SPREAD-SPECTRUM SIGNALS 780
F.J. Block and C.W. Baum

TRAINING SEQUENCE-BASED MULTIUSER CHANNEL ESTIMATION FOR BLOCK-SYNCHRONOUS CDMA 790
G. Caire and U. Mitra

SUPPRESSION OF HIGH-DENSITY, DYNAMIC NARROWBAND INTERFERENCE IN DS/CDMA SPREAD-SPECTRUM SYSTEM 800
C. Carlemalm, H.V. Poor, and A. Logothetis

LARGE SYSTEM PERFORMANCE OF REDUCED-RANK LINEAR FILTERS 810
M.L. Honig and W. Xiao

NONLINEAR MULTIUSER RECEIVERS WITH DISTRIBUTED POWER CONTROL IN CELLULAR RADIO NETWORKS 820
M.K. Varanasi

IV-A: WIRELESS COMMUNICATIONS II: SYSTEM CONSIDERATIONS IN PHYSICAL LAYER DESIGN

Organizers: V.V. Veeravalli and U. Madhow
(Cornell University and University of Illinois at Urbana-Champaign)
Chair: U. Madhow
(University of Illinois at Urbana-Champaign)

THE CODING-SPREADING TRADEOFF IN CDMA SYSTEMS 831
V.V. Veeravalli

SPECTRAL EFFICIENCY OF RANDOMLY SPREAD DS-CDMA IN A MULTI-CELL MODEL 841
B.M. Zaidel, S. Shamai, and S. Verdú

CDMA DESIGN THROUGH ASYMPTOTIC ANALYSIS: FADING CHANNELS 851
E. Biglieri, G. Caire, G. Taricco, and E. Viterbo

PACKING SPHERES IN THE GRASSMANN MANIFOLD: A GEOMETRIC APPROACH TO THE NON-COHERENT MULTI-ANTENNA CHANNEL 861
L. Zheng and D.N.C. Tse

BLIND ADAPTIVE MULTIUSER DETECTION FOR DS/SSMA COMMUNICATIONS WITH GENERALIZED RANDOM SPREADING IN A FREQUENCY-SELECTIVE FADING CHANNEL 871
J.H. Cho and J.S. Lehnert

TRAFFIC AIDED MULTIUSER DETECTION FOR PACKET SWITCHING RANDOM ACCESS/CDMA NETWORKS 881
B. Chen and L. Tong

IV-B: CODING THEORY IV

Chair: R.E. Blahut
(University of Illinois at Urbana-Champaign)

OPTICAL CHANNELS WITH DEAD TIME 891
A. Khandekar and R. McEliece

AWGN CODING THEOREMS FOR SERIAL TURBO CODES 893
H. Jin and R.J. McEliece

LINEAR CODES OVER Z/(2^K) OF CONSTANT EUCLIDEAN WEIGHT 895
J.A. Wood

SPACE-TIME TURBO CODES 897
Y. Liu and M.P. Fitz

TURBO DECODING OF CONCATENATED SPACE-TIME CODES 899
K.R. Narayanan

SOFT OUTPUT AND ITERATIVE STACK DECODING 901
R. Sivasankaran and S.W. McLaughlin

ALGEBRAIC GEOMETRIC CODES AND AN IMPROVEMENTS ON THE GILBERT-VARSHAMOV BOUND 903
H. Maharaj

IV-C: STOCHASTIC SYSTEMS AND CONTROL

Chair: G. Dullerud
(University of Illinois at Urbana-Champaign)

SINGLE-SAMPLE-PATH-BASED OPTIMIZATION OF MARKOV DECISION PROCESS 905
Z. Ren and B.H. Krogh

TRACKING CAPABILITY ANALYSIS OF THE LMS ALGORITHM FOR FIR SYSTEMS WITH AR COEFFICIENTS 915
Y. Wei, S.B. Gelfand, and J.V. Krogmeier

ADAPTIVE OPTIMAL PREDICTION FOR MIMO STOCHASTIC SYSTEMS USING CANONICAL FORMS 925
B. Shahrrava and J.D. Aplevich

SDRE ESTIMATOR PERFORMANCE IN A HARMONIC DISTORTION PROBLEM 935
E.E. Yaz, Y. Gao, and K.J. Olejniczak

STOCHASTIC DIFFERENCE EQUATIONS WITH TIME DELAYS 937
A.S.C. Sinha, S.E. Lyshevski, B.R. Pidaparti, and E. Kocaoglan

CONTROL UNDER COMMUNICATION CONSTRAINTS 940
S. Tatikonda and S. Mitter

IV-D: SPACE-TIME PROCESSING I

Organizer/Chair: A. Nehorai
(University of Illinois at Chicago)

SPACE-TIME FADING CHANNEL ESTIMATION IN UNKNOWN SPATIALLY CORRELATED NOISE 948
A. Dogandžić and A. Nehorai

BLAST TRAINING: ESTIMATING CHANNEL CHARACTERISTICS FOR HIGH CAPACITY SPACE-TIME WIRELESS 958
T.L. Marzetta

ANALYSIS OF THE PARALLEL INTERFERENCE CANCELER FOR DS/CDMA SIGNALS 967
R. Chandrasekaran and J.J. Shynk

BEARING ESTIMATION IN A RICEAN CHANNEL 977
G. Fuks, J. Goldberg, and H. Messer

OPTIMAL DOWNLINK BEAMFORMING USING SEMIDEFINITE OPTIMIZATION 987
M. Bengtsson and B. Ottersten

AN OVERVIEW OF A SIMULATION ENVIRONMENT TO STUDY THE IMPACT OF NON-IDEAL HARDWARE ON ARRAY PROCESSING 997
J. Yin, C.M.S. See, B.P. Ng, Y.K. Sin, and Y.L. Lu

SIGNAL REPRESENTATIONS FOR TRANSMIT-RECEIVE ANTENNA ARRAYS 1006
J. Zhang, K. Tantinarawat, and A.M. Sayeed

BLOCK SPACE-TIME ANTENNA PRECODING/DECODING FOR GENERALIZED MULTICARRIER COMMUNICATIONS IN UNKNOWN MULTIPATH 1016
Z. Liu, A. Scaglione, S. Barbarossa, and G.B. Giannakis

LEAST-SQUARES MULTI-USER FREQUENCY-DOMAIN CHANNEL ESTIMATION FOR BROADBAND WIRELESS COMMUNICATION SYSTEMS 1026
T.A. Thomas, F.W. Vook, and K.L. Baum

IV-E: MANUFACTURING SYSTEMS
Organizer/Chair: S. Reveliotis
(Georgia Institute of Technology)

A MARKOV DECISION PROCESS MODELING FOR CONTROL SWITCHING OF DISCRETE EVENT SYSTEMS 1036
H. Darabi and M.A. Jafari

SUPERVISORY CONTROL OF CONTRADICTIONS IN HIERARCHICAL TASK CONTROLLERS 1042
X. Guan and L.E. Holloway

IV-F: STOCHASTIC NETWORKS IV
Organizers: S.P. Meyn and R. Srikant
(University of Illinois at Urbana-Champaign)
Chair: R. Srikant
(University of Illinois at Urbana-Champaign)

MULTIUSER RECEIVERS, RANDOM MATRICES AND FREE PROBABILITY 1055
D.N.C. Tse

STABILITY PROPERTIES OF INCREMENTAL REDUNDANCY IN CDMA PACKET DATA NETWORKS 1065
R. Vijayakumar and K.M. Wasserman

PRICING PRIORITY CLASSES IN A DIFFERENTIATED SERVICES NETWORK 1075
P. Marbach

COMPARING TANDEM QUEUEING SYSTEMS AND THEIR FLUID LIMITS 1085
E. Altman, G. Koole, and T. Jiménez

WAITING TIME ASYMPTOTICS FOR TIME VARYING MULTISERVER QUEUES WITH ABANDONMENT AND RETRIALS 1095
A. Mandelbaum, W.A. Massey, M.I. Reiman, and A.L. Stolyar

LARGE DEVIATIONS FOR SMALL BUFFERS: AN INSENSITIVITY RESULT 1105
M. Mandjes and J.H. Kim

OVERFLOW AND LOSSES IN A NETWORK QUEUE WITH SELF-SIMILAR INPUT 1113
B. Tsybakov and N.D. Georganas

V-A: STOCHASTIC NETWORKS V
Organizers: S.P. Meyn and R. Srikant
(University of Illinois at Urbana-Champaign)
Chair: S.P. Meyn
(University of Illinois at Urbana-Champaign)

CHOKE A STATELESS MECHANISM FOR PROVIDING QUALITY OF SERVICE IN THE INTERNET 1122
R. Pan and B. Prabhakar

OPTIMAL ROUTING TO *M* PARALLEL QUEUES WITH NO BUFFERS 1132
E. Altman, S. Bhulai, B. Gaujal, and A. Hordijk

THE EFFECT OF SCALE ON INTERNET QUALITY 1142
M. Siler and J. Walrand

STABILITY OF MULTILANE INPUT-BUFFERED SWITCHES WITH MARKOV-MODULATED ARRIVAL PROCESSES 1152
P. Ho, D. Tse, and J. Walrand

V-B: WIRELESS COMMUNICATIONS III: NETWORKING ISSUES

Organizers: V.V. Veeravalli and U. Madhow
(Cornell University and University of Illinois at Urbana-Champaign)
Chair: V.V. Veeravalli
(University of Illinois at Urbana-Champaign)

ADAPTIVE TRANSMISSION FOR SPREAD-SPECTRUM COMMUNICATIONS OVER MULTIPATH CHANNELS 1162
M.B. Pursley and C.S. Wilkins

ADAPTIVE REDUNDANCY RETRANSMISSION PROTOCOLS FOR WIRELESS NETWORKS 1171
T. Ji and W.E. Stark

RESOURCE POOLING AND EFFECTIVE BANDWIDTHS FOR CDMA ANTENNA ARRAYS 1181
S.V. Hanly and D.N.C. Tse

ROUTING FOR MAXIMUM SYSTEM LIFETIME IN WIRELESS AD-HOC NETWORKS 1191
J.-H. Chang and L. Tassiulas

A SELF ORGANIZING WIRELESS SENSOR NETWORK 1201
K. Sohrabi, J. Gao, V. Ailawadhi, and G. Pottie

V-C: MULTIUSER DETECTION

Chair: M.K. Varanasi
(University of Colorado at Boulder)

BLIND ADAPTIVE NONCOHERENT MULTIUSER DETECTION FOR NONLINEAR MODULATION 1211
D. Das and M.K. Varanasi

LOW COMPLEXITY NON-COHERENT NEAR-OPTIMAL MULTIUSER DETECTION FOR OVERSATURATED MA COMMUNICATION 1221
R.E. Learned, A.S. Willsky, and D.M. Boroson

ADAPTIVE MULTIUSER DECISION FEEDBACK FOR ASYNCHRONOUS CELLULAR DS-CDMA 1236
R. Ratasuk, G. Woodward, and M.L. Honig

MULTIUSER EQUALIZATION FOR RANDOM SPREADING: LIMITS OF DECORRELATION WITH AND WITHOUT DECISION-FEEDBACK 1246
R.R. Müller

V-D: SPACE-TIME PROCESSING II
Organizer/Chair: A. Nehorai
(University of Illinois at Chicago)

SCHEDULING OF SWITCHED MULTIBEAM ANTENNAS IN A MULTIPLE ACCESS ENVIRONMENT........ 1256
A. Logothetis and H.V. Poor

MULTIPLE ANTENNA DIFFERENTIAL MODULATION........ 1266
B.M. Hochwald and W. Sweldens

SPACE-TIME ZERO FORCING EQUALIZATION FOR 3G CDMA FORWARD LINK TO RESTORE ORTHOGONALITY OF CHANNEL CODES........ 1274
M.D. Zoltowski and T.P. Krauss

A NOVEL SPACE-TIME SPREADING SCHEME FOR WIRELESS CDMA SYSTEMS........ 1284
B.M. Hochwald, T.L. Marzetta, and C.B. Papadias

V-E: CODING FOR MAGNETIC CHANNELS
Organizer/Chairs: E. Kurtas
(Quantrum Corporation)

2D EQUALIZATION FOR PAGE-ORIENTED DATA STORAGE SYSTEMS........ 1294
B.V.K.V. Kumar, V. Vadde, and M. Keskinoz

TOWARDS SOFT OUTPUT APP DECODING FOR NONSYSTEMATIC NONLINEAR BLOCK CODES........ 1304
K.D. Anim-Appiah and S.W. McLaughlin

LOW DENSITY PARITY CHECK CODES FOR MAGNETIC RECORDING........ 1314
J.L. Fan, A. Friedmann, E. Kurtas, and S. McLaughlin

DESIGN CONSIDERATIONS FOR CONCATENATING CONVOLUTIONAL CODES WITH PARTIAL RESPONSE CHANNELS........ 1324
W.E. Ryan

TURBO CODES FOR TWO-TRACK MAGNETIC RECORDING SYSTEMS........ 1334
E. Kurtas and T.M. Duman

LIST OF AUTHORS........ 1344

Representation and Expansion of (Max,Plus) Lyapunov Exponents

François Baccelli
ENS, DMI-LIENS, 45 rue d'Ulm, 75230 Paris Cedex 05, France.
Francois.Baccelli@ens.fr.

Stéphane Gaubert
INRIA, Domaine de Voluceau, 78153 Le Chesnay Cedex, France.
Stephane.Gaubert@inria.fr

Dohy Hong
ENS, DMI-LIENS, 45 rue d'Ulm, 75230 Paris Cedex 05, France.
Dohy.Hong@ens.fr

Abstract

We survey some recent representation and series expansions results for Lyapunov exponents of iterates of random monotone homogeneous maps. Under a memory-loss property, which holds generically in the special case of random products of matrices over the max-plus semiring, we give a perturbation series expansion formulas for the Lyapunov exponent, as a function of the probability law. The analyticity domain of the Lyapunov exponent is estimated through three different approaches: explicit Taylor series expansion, contraction property and language theory.

1 Introduction

We say that a map $f : \mathbb{R}^d \to \mathbb{R}^k$ is *monotone* if for all $X, Y \in \mathbb{R}^d$, $X \leq Y \implies f(X) \leq f(Y)$, where $\leq$ denotes the standard product ordering of $\mathbb{R}^d$. We say that f is (additively) *homogeneous* if for all $\lambda \in \mathbb{R}$ and $X \in \mathbb{R}^d$, $f(\lambda + X) = \lambda + f(X)$, where $\lambda + X$ denotes the vector $(\lambda + X_1, \dots, \lambda + X_d)$.

Given a finite set $\mathbb{S} = \{f_0, \dots, f_m\}$ of monotone homogeneous maps $\mathbb{R}^d \to \mathbb{R}^d$, we consider an infinite sequence $f_{i_1}, f_{i_2}, \dots$ of random elements of $\mathbb{S}$, taken under a stationary and ergodic probability law. Given a monotone homogeneous map $\varphi : \mathbb{R}^d \to \mathbb{R}$, we call *$\varphi$–Lyapunov exponent* the limit:

$$\gamma_\varphi = \lim_{n\to\infty} \frac{1}{n} \mathbb{E}[\varphi \circ f_{i_n} \circ \cdots \circ f_{i_1}(X)], \tag{1}$$

where X is an arbitrary point in $\mathbb{R}^d$ (it is easy to see that if γ_φ exists for a $X \in \mathbb{R}^d$, then it exists for all integrable initial conditions and it has the same value).
Intuitively, the φ–Lyapunov exponent measures the growth rate of $\varphi(X_n)$, where

$$X_n = f_{i_n} \circ \cdots \circ f_{i_1}(X), \ \ X_0 = X.$$

For instance, setting $\mathrm{t}X \overset{\text{def}}{=} \max_{1\leq i\leq d}(X)_i$ ("t" stands for "top"), we obtain γ_{t}, a quantity which is sometimes called *top*–Lyapunov exponent. Dually, setting $\mathrm{b}X \overset{\text{def}}{=} \min_{1\leq i\leq d}(X)_i$, we obtain the *bottom*–Lyapunov exponent.

Indeed, we shall see in §2 that in some good cases, the limit (1) is independent of φ, and that also an almost sure convergence result holds.

Lyapunov exponents of non-negative matrices and (max, +) Lyapunov exponents, which are notoriously difficult to compute or approximate [14], are special cases of Lyapunov exponents of monotone homogeneous maps.

In this paper, our purpose is to expand γ_φ as a power series in the parameter of the probability law and to determine the convergence radius of this series or more generally to estimate the analyticity domain of γ_φ.

On $\mathbb{R}^d$ we define the equivalence relation: $X \sim Y$ if $\exists\lambda \in \mathbb{R}$ such that $X = Y + \lambda$. The crucial assumption of this paper is the existence of at least one map, $f = f_{j_k}\circ\cdots\circ f_{j_1}$ which is a finite iteration of elements of $\mathbb{S}$, whose image is a *line* i.e. $\forall X, Y \in \mathbb{R}^d, f(X) \sim f(Y)$. There are two reasons to make such an assumption: the first one comes from linear maps over max-plus semiring where this property occurs generically, in particular for the Lyapunov exponent associated with random heaps of pieces (see [9], [6] and examples §2). The second reason is that this is the only cases when an explicit series expansion of the Lyapunov exponents is known [4].

2 Existence of Lyapunov exponent and examples

We define $\mathbb{S}^k = \{f_{j_k}\circ\cdots\circ f_{j_1} \mid f_{j_l} \in \mathbb{S}, l = 1,.., k\}$. We denote by $\mathbb{S}^+$ the semi-group generated by $\mathbb{S}$. We consider a stationary ergodic sequence $\{f_{i_n}\}_{n\in\mathbb{Z}}$ of elements of $\mathbb{S}$. With an element $g = f_{j_k}\circ\cdots\circ f_{j_1} \in \mathbb{S}^+$, we associate the time $T_n(g)$:

$$\begin{aligned} T_0(g) &= -\inf\{m \geq 0 \mid f_{i_{-m}}\circ\cdots\circ f_{i_{-m-k}} = g\}, \\ T_n(g) &= \inf\{m > T_{n-1} \mid f_{i_m}\circ\cdots\circ f_{i_{m-k}} = g\}, \text{ for } n \geq 1. \end{aligned}$$

We say that the sequence $\{f_{i_n}\}$ has the *memory loss property* if there is a $g \in \mathbb{S}^+$, whose image is a line and such that if $g \in \mathbb{S}^k$, $P(f_{i_k}\circ\cdots\circ f_{i_1} = g) > 0$. We denote by $\mathbb{F}$ the set of such g. Without loss of generality, we will assume that if $f_{j_k}\circ\cdots\circ f_{j_1} \in \mathbb{F}$, $\mathbb{F}$ does not contain any strict sub-element $f_{j_l}\circ\cdots\circ f_{j_{l'}}$, $k \geq l \geq l' \geq 1$ of $f_{j_k}\circ\cdots\circ f_{j_1}$.

Lemma 1. *Assume that $\{f_{i_n}\}$ is a stationary and ergodic sequence of random monotone and homogeneous maps with memory loss property. Then the following limits exist regardless of the initial condition:*

$$\lim_{n\to+\infty} \frac{X_n}{n} = \begin{bmatrix} \gamma \\ \vdots \\ \gamma \end{bmatrix} = \Gamma \quad a.s. \text{ and in } L_1. \tag{2}$$

As a consequence, for all monotone homogeneous maps φ,

$$\gamma_\varphi = \gamma.$$

Proof. Using Kingman's sub-additive ergodic theory [10], [7], [15] we can show classically that the following limits exist:

$$\lim_{n\to\infty} \frac{\mathrm{t}X_n}{n} = \gamma_{\mathrm{t}}, \quad \lim_{n\to\infty} \frac{\mathrm{b}X_n}{n} = \gamma_{\mathrm{b}} \text{ a.s. and in } L_1.$$

If $g \in \mathbb{F}$, we consider T_n associated with g. Then $\mathtt{t}X_{T_n} - \mathtt{b}X_{T_n} = \mathtt{t}g(0) - \mathtt{b}g(0)$. Therefore $\frac{\mathtt{t}X_{T_n} - \mathtt{b}X_{T_n}}{T_n} \to 0$, when $n \to +\infty$. Since for all $n \geq 0$, T_n is a.s. finite (from ergodicity), this proves that there exists a subsequence of $\frac{\mathtt{t}X_n - \mathtt{b}X_n}{n}$ which tends to 0. But $\frac{\mathtt{t}X_n - \mathtt{b}X_n}{n} \to \gamma_{\mathtt{t}} - \gamma_{\mathtt{b}}$ a.s. It follows that $\gamma_{\mathtt{t}} = \gamma_{\mathtt{b}}$, and for all monotone homogeneous map φ, $\varphi(0) + \mathtt{b}X_n \leq \varphi(X_n) \leq \varphi(0) + \mathtt{t}X_n$, which proves the last assertion. $\square$

Lemma 1 still holds, if, in the foregoing assumptions, one replace the property *the image of g is a line*, by *the image of g is compact.*

Example 1 (Tetris model). We assume that $\{f_{i_n}\}_{n\in\mathbb{N}}$ is an i.i.d. sequence of $\mathbb{S} = \{f_0, f_1, f_2\}$, such that $p_j = P(f_{i_n} = f_j)$ with $p_0 + p_1 + p_2 = 1$, and where $f_0, f_1, f_2 : \mathbb{R}^2 \to \mathbb{R}^2$ are given by

$$f_1(x,y) = (x+1, y),$$
$$f_2(x,y) = (x, y+1),$$
$$f_0(x,y) = (\max(x,y)+1, \max(x,y)+1).$$

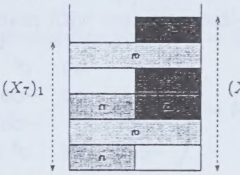


Figure 1

The action of the maps f_j can be interpreted graphically, as shown in Figure 1. It is not difficult to see that $(X_n)_i = (f_{j_n} \circ \cdots \circ f_{j_1}(0))_i$ represents the height of column i of a heap of pieces, built by piling up the pieces corresponding to f_{j_1}, ..., f_{j_n}, in this order (for more details see [9]). In this case, $f_0 \in \mathbb{F}$ and the Lyapunov exponent $\gamma(p)$ was calculated explicitly in [6] :

$$\gamma(p_1, p_2) \quad = \quad \frac{1}{2}\frac{p_1 + p_2 - 4p_1p_2 + (2 - p_1 - p_2)\sqrt{1 - 4p_1p_2}}{\sqrt{1 - 4p_1p_2}}. \tag{3}$$

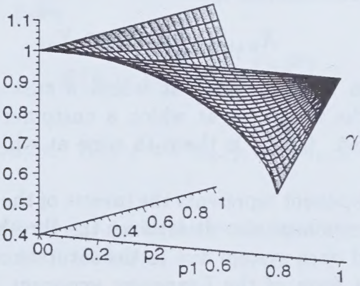
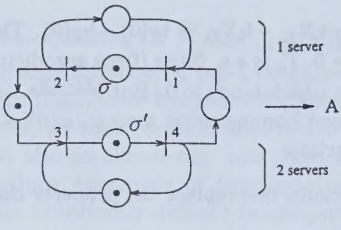


Figure 2

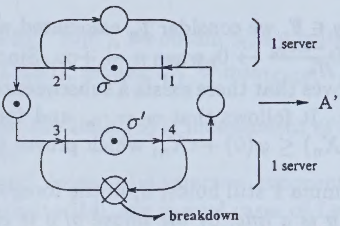
We see that γ is analytic in the domain $\{|p_1| + |p_2| < 1\}$ with a singularity at $p_1 = p_2 = \frac{1}{2}$.

Example 2 (Stochastic network). Consider a closed FIFO network with two stations, where the first one is a single server station, and the second one has two servers, one of which may break down (see the Petri net description below). Each customer arriving at station 1 demands a service time σ and each customer arriving at station 2 a service time σ'. When the n-th customer enters station 2 ($n \geq 1$),

- either there is a breakdown (which takes place with probability p) and the customer must wait for the server to be available;
- or there is no breakdown (which takes place with probability $1-p$) and the customer must wait for one of two servers to be available.



Main state described by A



Breakdown case described by A'

As detailed in [4], the evolution of this network can be described as a stochastic (max,+)–linear system with matrices A and A' given by

$$A = \begin{pmatrix} \sigma & -\infty & \sigma' & -\infty \\ \sigma & -\infty & -\infty & -\infty \\ -\infty & 0 & -\infty & 0 \\ -\infty & -\infty & \sigma' & -\infty \end{pmatrix}, \quad A' = \begin{pmatrix} \sigma & -\infty & \sigma' & -\infty \\ \sigma & -\infty & -\infty & -\infty \\ -\infty & 0 & \sigma' & -\infty \\ -\infty & -\infty & \sigma' & -\infty \end{pmatrix}.$$

We recall that the (max, +) product $A \otimes X$ of a matrix $A \in (\mathbb{R} \cup \{-\infty\})^{d \times d}$ by a vector $X \in (\mathbb{R} \cup \{-\infty\})^d$ is defined by

$$(A \otimes X)_i = \max_{1 \le j \le d} A_{ij} + (X)_j.$$

When A has at least one finite entry per row, the map $X \to A \otimes X$ is monotone and homogeneous $\mathbb{R}^d \to \mathbb{R}^d$.

Suppose that $A(n)$ is i.i.d. and equal to A or A' with probability $1-p$ and p respectively. Then the stochastic recurrence,

$$X_{n+1} = A(n+1) \otimes X_n,$$

is such that $(X_n)_i$ gives the n-th time at which a customer leaves transition i, i.e. $(X_n)_1$(resp. $(X_n)_3$) is the n-th time at which a customer starts its service in station 1(resp. 2) and $(X_n)_2$(resp. $(X_n)_4$) is the n-th time at which a customer leaves station 1(resp. 2).

Here the Lyapunov exponent represents the inverse of the stationary throughput of the system. The Lyapunov exponent also determines the threshold which gives the stability domain of the associated open system due to the saturation rule [3].

Although an explicit form of the Lyapunov exponent is not known, the following results gives its series representation together with its analyticity domain.

3 Representation formulas

The results given in this section are essentially based on the regenerative property of X_n. Let $\{T_n\}_{n \in \mathbb{Z}}$ be the sequence of times at which elements of $\mathbb{F}$ appear in the sequence $\{f_{i_n}\}_{n \in \mathbb{Z}}$ i.e.

$$\begin{aligned} T_0 &= \sup\{k \le 0 : \exists m \ge 0 \text{ s.t. } f_{i_k} \circ \cdots \circ f_{i_{k-m}} \in \mathbb{F}\}, \\ T_n &= \inf\{k > T_{n-1} : \exists m \ge 0 \text{ s.t. } f_{i_k} \circ f_{i_{k-m}} \in \mathbb{F}\}, \quad \text{if } n \ge 1, \\ &= \sup\{k < T_{n+1} : \exists m \ge 0 \text{ s.t. } f_{i_k} \circ f_{i_{k-m}} \in \mathbb{F}\}, \quad \text{if } n < 0. \end{aligned}$$

If the sequence $\{f_{i_n}\}_{n\in\mathbb{Z}}$ satisfies the memory loss property, it follows from its definition that T_n is a.s. finite for all n.
Define

$$T'_n \;=\; \inf\{m \geq 0 \text{ s.t. } f_{i_{T_n}} \circ \cdots \circ f_{i_{T_n-m}} \in \mathbb{F}\}.$$

Theorem 1 (Cycle formula). *If the sequence $\{f_{i_n}\}_{n\in\mathbb{Z}}$ is stationary such that $\exists g \in \mathbb{F}$, $\mathbb{E}[T_0(g)] < +\infty$ then we have*

$$\mathbb{E}[\overline{X}_1 - \overline{X}_0] \;=\; \frac{\mathbb{E}^0[\overline{X}_{T_1} - \overline{X}_0]}{\mathbb{E}^0[T_1]}, \tag{4}$$

where $\mathbb{E}^0$ is the expectation w.r.t. the conditional probability given $T_0 = 0$, and $\overline{X}_0 = f_{i_0} \circ \cdots \circ f_{i_{T_0 - T'_0}}(0)$, $\overline{X}_k = f_{i_k} \circ \cdots \circ f_{i_1}\overline{X}_0$.

Proof. We put

$$f = \lim_{n\to\infty} f_{i_1} \circ \cdots \circ f_{i_{-n}}(0) - f_{i_0} \circ \cdots \circ f_{i_{-n}}(0).$$

Since $f = f_{i_1} \circ \cdots \circ f_{i_{T_0 - T'_0}}(0) - f_{i_0} \circ \cdots \circ f_{i_{T_0-T'_0}}(0)$, from the monotonicity and homogeneity of f_i we obtain,

$$\min_{i,j}(f_i(0))_j \times (T'_0 - T_0 + 1) \leq f_{i_0} \circ \cdots \circ f_{i_{T_0 - T'_0}}(0) \leq \max_{i,j}(f_i(0))_j \times (T'_0 - T_0 + 1),$$

and thus $\exists K, K' \in \mathbb{R}$ s.t.

$$|f| \leq K(|T'_0| + |T_0|) + K'.$$

If $g \in \mathbb{S}^k$, then $|T'_0| + |T_0| \leq |T_0(g)| + k$. Therefore f is integrable. Then from the Palm inversion formula (see [1], p.44)

$$\mathbb{E}[f] = \frac{\mathbb{E}^0[\sum_{n=0}^{T_1-1} f \circ \theta^n]}{\mathbb{E}^0[T_1]},$$

where θ is the shift operator. But $f \circ \theta^n \mathbf{1}_{n<T_1} = \overline{X}_{n+1} - \overline{X}_n$ and $\sum_{n=0}^{T_1-1} f \circ \theta^n = \overline{X}_{T_1} - \overline{X}_0$. □

Corollary 1. *If the sequence $\{f_{i_n}\}_{n\in\mathbb{Z}}$ is stationary and ergodic and if there exists $g \in \mathbb{F}$ such that $\mathbb{E}[T_0(g)] < +\infty$, then we have*

$$\Gamma = \mathbb{E}[\overline{X}_1 - \overline{X}_0].$$

Proof. From the ergodic theorem :

$$\frac{1}{m} \sum_{0\leq k\leq m-1} [X_{k+1} - X_k] \xrightarrow{a.s.} \frac{\mathbb{E}\left[X_{T_2} - X_{T_1}\right]}{\mathbb{E}\left[T_2 - T_1\right]} = \frac{\mathbb{E}^0\left[X_{T_1} - X_0\right]}{\mathbb{E}^0\left[T_1\right]}$$

and

$$\frac{1}{m} \sum_{0\leq k\leq m-1} [X_{k+1} - X_k] = \frac{X_m}{m} \xrightarrow{a.s.} \Gamma.$$

□

4 Simple Bernoulli case

In this section, $\{f_{i_n}\}_{n\in\mathbb{N}}$ is an i.i.d. sequence of $\mathbb{S} = \{f_0, f_1\}$ such that for all n,

$$f_{i_n} = \begin{cases} f_0 & \text{with probability } p_0 = 1-p, \\ f_1 & \text{with probability } p_1 = p. \end{cases}$$

Moreover we will assume that $\exists c \in \mathbb{N}$ such that $f_0^c \in \mathbb{F}$.

4.1 Series expansion formulas

Theorem 2 (Taylor series expansion). *In a neighbourhood of* 0,

$$\Gamma(p) = \sum_{l\geq 0} \Pi_l p^l,$$

with

$$\Pi_l = \begin{bmatrix} \pi(l) \\ \vdots \\ \pi(l) \end{bmatrix} = (-1)^l \left\{ \binom{c}{l-1} f_0{}^c(0) + \binom{c+1}{l} \Gamma(0) \right\} +$$

$$\sum_{k=1}^{l} \left\{ (-1)^{l-k} \sum_{j_1,..,j_{k-1}=0}^{c-1} \binom{2c+j_1+..+j_{k-1}}{l-k} f_0{}^c \circ f_1 \circ f_0{}^{j_1} \circ f_1 \cdots f_0{}^{j_{k-1}} \circ f_1 \circ f_0{}^c(0) \right\}.$$

Proof. The formula for $\pi(l)$ is based on combinatorial arguments [4]. Nevertheless the form of the coefficients of the expansion can be forseen from the cycle formula (4): the expansion of $\mathbb{E}^0[\overline{X}_{T_1}]$ shows that π_l is necessarily a linear combination of $f_0{}^c \circ f_1 \circ f_0{}^{j_1} \circ f_1 \cdots f_0{}^{j_{k-1}} \circ f_1 \circ f_0{}^c(0)$. □

The computation cost of these coefficients is exponential: indeed without the assumption on $\mathbb{F}$, it is proved that [14] the approximation of γ may be a NP-hard problem (for instance in (max, +)-semiring) and even γ can be algorithmically not approximable (in $(\mathbb{R}, +, \times)$).

Application From the Taylor series formula of Theorem 2, we find again the expression (3) of Example 1. The above formula also gives the Taylor series expansion of the Lyapunov exponent of Example 2 when $\sigma' < 2\sigma$: we check that $c = 4$, $\gamma(0) = \sigma$.

1. If $2\sigma' < 3\sigma$, $\gamma(p) = \sigma$.
2. If $2\sigma' > 3\sigma$, we find by direct computations the following series expansion :

$$\begin{aligned} \gamma(p) &= \sigma + (2\sigma' - 3\sigma)p + (5\sigma - 3\sigma')p^2 + (4\sigma' - 6\sigma)p^3 + (7\sigma - 5\sigma')p^4 \\ &\quad + (9\sigma' - 13\sigma)p^5 + (26\sigma - 17\sigma')p^6 + \cdots \end{aligned}$$

Theorem 3 (Cycle formula). *If* $p < 1$,

$$\begin{aligned} \Gamma(p) &= \lim_{n\to\infty} \mathbb{E}[X_{n+1} - X_n] \\ &= \sum_{f\in\mathcal{S}_e} \mathbb{E}\left[X_1 - X_0 \mid X_0 = \hat{f}(X)\right] p(f), \qquad (5) \\ &= \sum_{k\geq c} \sum_{f\in\mathcal{S}_e, |f|=k} \mathbb{E}\left[X_1 - X_0 \mid X_0 = \hat{f}(X)\right] p(f). \qquad (6) \end{aligned}$$

where $\mathcal{S}_e = \{(f_{j_n}, \cdots, f_{j_1}, f_1, f_0^c) \mid f_{j_n} \circ \cdots \circ f_{j_1}$ *does not contain* f_0^c, $n \in \mathbb{N}\} \cup \{f_0^c\}$ *and where* $|f|$ *is the length* n, $p(f) = p_{j_n} \times \cdots \times p_{j_1}$, $\hat{f} = f_{j_n} \circ \cdots \circ f_{j_1}$ *if* $f = (f_{j_n}, \cdots, f_{j_1})$. *Besides,* $\exists K, K' > 0$, *such that* $\forall n \geq c$,

$$\sum_{k \geq n} \sum_{f \in \mathcal{S}_e, |f| = k} ||\mathbb{E}[X_1 - X_0 \mid X_0 = \hat{f}(X)]||_\infty p(f) \leq (Kn + K')\alpha^n. \tag{7}$$

where α is the inverse of the first non-negative root of the equation

$$p_1 x \sum_{i=0}^{c-1} (p_0 x)^i = 1.$$

Proof. A proof of the first part is given in [8]. The representation (5) is in fact a direct application of Corollary 1. The bound (7) can be obtained from the counting automaton which recognize the first occurrence f_0^c: α^{-1} is the first non-negative pole of the generating function of $T_0(f_0^c)$. □

Notice that α gives the exact exponential convergence rate of the residue and that we have the bound:

$$\alpha \leq (1 - p_0^c)^{\frac{1}{c}},$$

since, independently for each $j = 1, .., \left[\frac{n}{c}\right]$, the probability that $(f_0, \cdots, f_0)$ does not appear in the subsequence $(f_{i_{jc}}, .., f_{i_{(j-1)c+1}})$ of length c is $1 - p_0^c$. The series expansion (6) holds for all $p_0 > 0$ and is more efficient than the Taylor series expansion. Yet its computation depends on the probability p_0, while the coefficients of the Taylor series depend only on f_0 and f_1.

This last representation also gives an accurate estimate of the analyticity domain, as we will see below.

4.2 Analyticity region

The basic argument to estimate the analyticity region uses the fact that γ is analytic when the following uniform convergence holds:

$$\Gamma(p) = \lim_{n \to +\infty} \mathbb{E}[X_n - X_{n-1}].$$

Indeed, $\mathbb{E}[X_n - X_{n-1}]$ is polynomial in p, and the uniform limit of analytic function is analytic.

Direct bound: From the expression of π_l, we can get a first estimate of the radius of convergence of $\gamma(p)$ at point 0 ($RC(\gamma)$) together with a bound on the error of the expansion:

Corollary 2 (Analyticity region, first estimate). *We have*

$$RC(\gamma) \geq \frac{1}{2c},$$

and for all $p \in [0, \delta]$, *with* $\delta < \frac{1}{2c}$, *the error term in the Taylor expansion of* $\gamma(p)$ *of order* l *is bounded by:*

$$\frac{D(2c\delta)^{l+1}[1 + l(1 - 2c\delta)]}{(1 - 2c\delta)^2},$$

where $D = 2\left[\mathrm{t}|f_0(0)| \vee \mathrm{t}|f_1(0)| + \mathrm{t}|{f_0}^c(x)|\right](c+1)$.

Proof. See [4]. □

Contraction property: The contraction approach is an alternative method which has been considered by D. Ruelle [13] and Y. Peres [12] for conventional product of matrices.

Monotone homogeneous maps $\mathbb{R}^d \to \mathbb{R}^d$ are non-expansive for the additive version $||.||_\mathcal{D}$ of Hilbert's projective metric, i.e.:

$$\tau(f) = \sup_{X,Y\in\mathbb{R}^d} \left\{ \frac{||f(X)-f(Y)||_\mathcal{D}}{||X-Y||_\mathcal{D}}, ||X-Y||_\mathcal{D} \neq 0 \right\} \leq 1,$$

where $||X||_\mathcal{D} = \mathrm{t}X - \mathrm{b}X$. This property is fundamental to the proof of the following theorem.

Theorem 4 (Contraction region, estimate 2). *The Lyapunov exponent can be extended analytically on the domain*

$$\{(p_0,p_1) \in \mathbb{C}^2 \text{ s.t. } \tau(f_0)|p_0| + \tau(f_1)|p_1| < 1\}. \tag{8}$$

In fact, the region given by (8) holds more generally, even if $\mathbb{F}$ is empty, for monotone homogeneous maps [5].

Corollary 3. *We have*

$$RC(\gamma) \geq x,$$

where x is the first positive root of the equation

$$(1+2x)^c - (1+x)^c = 1.$$

Proof. It is easy to see that $\tau(f)$ is equal to 0, if $f \in \mathbb{F}$. Then put $Y_n = X_{nc}$, and apply multinomial version of Theorem 4 on Y_n. □

For $c \leq 15$, this result is sharper than the one obtained by direct bounds on the coefficients of the series expansion. Yet this is not satisfying enough, since the contraction region are usually strictly included in the analyticity region.

Cycle formula: This approach gives the optimal value for $RC(\gamma)$.

Theorem 5 (Analyticity region, estimate 3). *On the open domain*

$$|p|(1+|1-p|+\cdots+|1-p|^{c-1}) < 1, \tag{9}$$

Γ can be extended to an analytic function of p.

Proof. It is not difficult to see that the absolute convergence domain of the series (5) coincides with the absolute convergence domain of

$$Z = \sum_{f\in S_c} p(f).$$

This kind of partition domain can be computed quite easily: here $Z = \frac{p_0^c}{1-p_1(\sum_{k=0}^{c-1} p_0^k)}$. More generally, when p is a rational probability and $\mathbb{F}$ a rational set of sequences, using language theory arguments [8]. □

Corollary 4. *We have*

$$RC(\gamma) \geq 2^{\frac{1}{c}} - 1. \tag{10}$$

Under the assumption ${f_0}^c \in \mathbb{F}$, an example of Lyapunov exponent which has a pole at $p = -(2^{\frac{1}{c}} - 1)$ is given in [8]: this implies that one can not improve the bound (10) under that assumption.

Application In Example 2, $RC(\gamma) \geq 2^{\frac{1}{4}} - 1 = 0.189207115$. But one may have $RC(\gamma) > 2^{\frac{1}{c}} - 1$: Figure 3 suggests that $RC(\gamma)$ may be near 0.4.

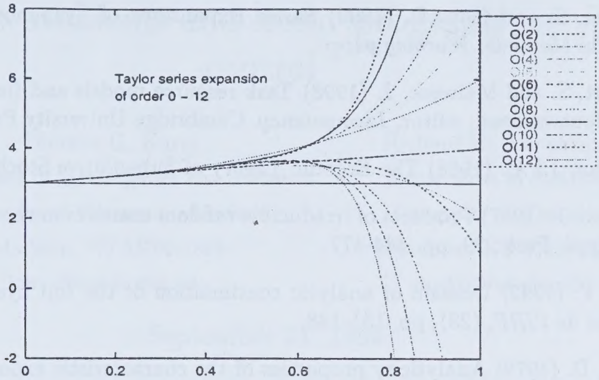


Figure 3: $\sigma = 3$, $\sigma' = 5$.

5 Conclusion

In this paper we presented the simplest version of results on the Lyapunov exponent under the memory loss property with very brief ideas of proof. Their generalizations (to multinomial scheme or to Markov dependent probability or to rational set $\mathbb{F}$ etc.) can be found in [4], [5] and [8]. It would be interesting to obtain similar results relaxing the memory loss assumption.

References

[1] Baccelli, F., Brémaud, P. (1994) Elements of Queueing Theory. *Springer-Verlag.*

[2] Baccelli, F., Cohen, G. Olsder, G.J. and Quadrat, J.P. (1992) Synchronization and Linearity, Wiley.

[3] Baccelli, F., Foss, S. (1995) On the saturation rule for the stability of queues. *J. Appl. Prob.*, 32, pp. 494-507.

[4] Baccelli, F. and Hong, D. (1998) Analytic Expansions of (max,+) Lyapunov Exponents. *INRIA Report*, May, *N*° 3427, INRIA Sophia-Antipolis, *To appear in Annals of Appl. Prob.*

[5] Baccelli, F. and Hong, D. (1998) Analyticity of Iterates of Random Non-Expansive Maps. *INRIA Report*, November, *N*° 3558, INRIA Sophia-Antipolis, *To appear in Adv. Appl. Prob. in the issue of March 2000.*

[6] Brilman, M. and Vincent, J.M. (1997) Dynamics of synchronized parallel systems, *Stochastic Models*, 3, 13, pp. 605-618.

[7] Cohen, J.E. (1988) Subadditivity, generalized product of random matrices and operations research. *SIAM Review*, 30(1) : pp. 69-86.

[8] Gaubert, S. and Hong,D. (1999) Series Expansions of Lyapunov Exponents and Forgetful Monoids. *Working paper.*

[9] Gaubert, S. and Mairesse, J. (1998) Task resource models and (max,+) automata. In J. Gunawardena, editor, *Idempotency.* Cambridge University Press.

[10] Kingman, J.F.C. (1968) The Ergodic Theory of Subadditive Stochastic Processes.

[11] Mairesse, J. (1997) Products of irreducible random matrices in the (max,+) algebra. *Adv. Appl. Prob.*, 29, pp. 444-477.

[12] Peres, Y. (1992) Domain of analytic continuation of the top Lyapunov exponent. *Annales de l'IHP*, (28), pp. 131-148.

[13] Ruelle, D. (1979) Analyticity properties of the characteristic exponents of Random matrix products. *Adv. Math.*, (32), pp. 68-80.

[14] Tsitsiklis, J. and Blondel, V. (1997) When is a pair of matrices mortal? *Information Processing Letters*, 63, pp. 283-286.

[15] Vincent, J.M. (1997) Some ergodic results on stochastic iterative discrete event systems. *Journal of Discrete Event Dynamic Systems*, (7,2) pp. 209-232.

Martingale problems and linear programs for singular control *

Thomas G. Kurtz
Departments of Mathematics and Statistics
University of Wisconsin - Madison
Madison, WI 53706-1388
kurtz@math.wisc.edu

Richard H. Stockbridge
Department of Statistics
University of Kentucky
Lexington, KY 40506-0027
stockb@ms.uky.edu

September 21, 1999

Abstract

Using a controlled Jackson network as the primary example, stochastic control problems are formulated as martingale problems and the optimal solutions are characterized as the solutions of infinite dimensional linear programs. Assuming a heavy traffic scaling, the analogous linear program for a singular control problem is derived, and the solution of the linear program is shown to give the stochastic model minimizing the long-run average cost.

1 The model.

We consider a controlled Jackson network with m stations, external arrival rates $v_1, \ldots, v_m$ service rates $u_1, \ldots, u_m$, and routing probabilities p_{ij}. We assume that the routing probabilities are fixed but that we can control the arrival and service rates subject, perhaps, to certain constraints.

We can formulate the model as the solution of a system of stochastic equations:

$$Q(t) = Q(0) + Y_i(\int_0^t V_i(s)ds) + \sum_{j=1}^m Y_{ji}(p_{ji}\int_0^t I_{\{Q_j(s)>0\}}U_j(s)ds)$$
$$- \sum_{j=1}^m Y_{ij}(p_{ij}\int_0^t I_{\{Q_i(s)>0\}}U_i(s)ds),$$

where $\{Y_i\}$ and $\{Y_{ij}\}$ are independent, unit Poisson processes, $\{V_i\}$ are the controlled arrival rates, and $\{U_i\}$ are the controlled service rates.

The generator of the model has the form

$$\begin{aligned} Af(q,u,v) &= \sum_{i=1}^m v_i(f(q+e_i) - f(q)) + \sum_{i=1}^m u_i p_{i0} I_{\{q_i>0\}}(f(q-e_i) - f(q)) \\ &\quad + \sum_{i,j=1}^m u_i p_{ij} I_{\{q_i>0\}}(f(q+e_j-e_i) - f(q)), \end{aligned}$$

---

*This research is supported in part by the National Science Foundation

where, for $i = 1, \ldots, m$, e_i denotes the vector with ith component 1 and all others 0. Take the domain of A, $\mathcal{D}(A)$, to be the collection of bounded functions on $\mathbb{Z}_0^m$, $\mathbb{Z}_0 = \{0, 1, 2, \ldots\}$, and assume that there is a constant such that the controlled rates must satisfy $U_i, V_i \leq K$. Then, assuming the U_i and V_i are nonanticipating in an appropriate sense, any solution of the system of equations has the property that for every $f \in \mathcal{D}(A)$,

$$f(Q(t)) - \int_0^t Af(Q(s), U(s), V(s))ds \tag{1.1}$$

is a martingale with respect to a filtration independent of f.

We require

$$0 \leq v_i \leq K, 0 \leq u_i$$

$$\sum_i \alpha_{ki} u_i \leq 1, \quad k = 1, \ldots, k_0 \tag{1.2}$$

and

$$\sum_i \beta_{li} v_i \geq 1, \quad l = 1, \ldots, l_0, \tag{1.3}$$

where $\alpha_{ki}, \beta_{li} \geq 0$. We assume that for each i, there exists at least one k such that $\alpha_{ki} > 0$, and, hence, $u_i \leq 1/\alpha_{ki}$. Let $P = ((p_{ij}))_{1 \leq i,j \leq m}$ be the matrix of transition probabilities where p_{ij} is the probability that a customer of class i moves to class j after completing service. $p_{i0} = 1 - \sum_{j=1}^m p_{ij}$ is the probability that the customer leaves the system. We assume that the system is open so that

$$(I - P)^{-1} = \sum_{k=0}^{\infty} P^k$$

exists.

Let $\mathbf{1}$ be the vector all of whose components are ones. We can write the constraints in (1.2) and (1.3) as $\alpha u \leq \mathbf{1}$ and $\beta v \geq \mathbf{1}$, where α is the matrix with rows $\alpha_k = (\alpha_{k1}, \ldots, \alpha_{km})$, $k = 1, \ldots, k_0$, and β is the matrix with rows $\beta_l = (\beta_{l1}, \ldots, \beta_{lm})$. Define

$$\Gamma_\beta = \{v \in [0, K]^m : \beta v \geq \mathbf{1}\}, \quad \Gamma^\alpha = \{u \in [0, \infty)^m : \alpha u \leq \mathbf{1}\},$$

that is, Γ_β is the collection of admissible external arrival rates and Γ^α is the collection of admissible service allocations.

In some examples, it may be more natural to state arrival constraints as equalities rather than inequalities. For example, the external arrival rate into a queue may be fixed. Ordinarily, however, replacing some of the inequalities by equalities will not affect the analysis.

This model includes the Markov version of the model considered by Laws [8]. In his model, job types have fixed routes, so p_{ij} is 0 or 1, but models with more general routing probabilities can be reformulated as models with fixed routes. In his model, a finite collection of servers allocate effort among disjoint collections J_k of job classes, which, in our notation, gives constraints of the form

$$\sum_{i \in J_k} \frac{u_i}{\mu_i} \leq 1, \quad k = 1, \ldots, k_0,$$

that is, for $i \in J_k$, u_i/μ_i is the fraction of the effort of server k that is devoted to job class i.

The model also includes special cases of the Markov, continuous time analog of the model considered by Martins and Kushner [10]. Some of their examples allow control of the $\{p_{ij}\}$.

1.1 Example: Two queues with alternate routing.

(See [10].) Assume that there are two queues that can serve customers at maximum rates μ_1 and μ_2 respectively. There are three arrival streams with rates κ_1, κ_2, and κ_3. All traffic from the first stream enters queue I, all traffic from the second stream enters queue II, and the traffic from the third stream can be routed to either queue. The constraints become

$$v_1 \geq \kappa_1, \quad v_2 \geq \kappa_2, \quad v_1 + v_2 \geq \kappa_1 + \kappa_2 + \kappa_3$$

and

$$u_1 \leq \mu_2, \qquad u_2 \leq \mu_1.$$

Of course, these inequalities can be normalized to be of the form above by dividing through by the right side.

1.2 Example: Two queues with reassignable server.

(See [1, 4, 12].) Assume that there are two queues with arrival rates κ_1 and κ_2 and two servers. The first server can only serve customers in the first queue and serves these customers at rate μ_1. The second server can serve customers in the second queue at rate μ_2 and/or customers in the first queue at rate μ_3. More precisely, if the second server allocates a fraction θ of its effort to the second queue and the remaining fraction $(1-\theta)$ to the first queue, then it serves customers in the second queue at rate $\theta\mu_2$ and customers in the first queue at rate $(1-\theta)\mu_3$. The constraints then become

$$v_1 \geq \kappa_1, \qquad v_2 \geq \kappa_2$$

and

$$u_1 \leq \mu_1 + \mu_3, \quad u_2 \leq \mu_2, \quad \frac{u_1}{\mu_3} + \frac{u_2}{\mu_2} \leq \frac{\mu_1}{\mu_3} + 1.$$

Again, the inequalities can be normalized by dividing through by the right side.

1.3 Example: Two types of work and two servers.

Assume that there are two types of work that arrive in the system at rates κ_1 and κ_2. There are two servers. The first server can process Type 1 work at rate μ_1 and Type 2 work at rate μ_2. The second server can process Type 1 work at rate μ_3 and Type 2 work at rate μ_4. The work can by allocated to the servers in any manner. The system has four customer classes and the corresponding constraints are

$$v_1 + v_3 \geq \kappa_1, \qquad v_2 + v_4 \geq \kappa_2$$
$$\frac{u_1}{\mu_1} + \frac{u_2}{\mu_2} \leq 1, \qquad \frac{u_3}{\mu_3} + \frac{u_4}{\mu_4} \leq 1\,.$$

2 Conditions for stability.

Let $\lambda_j(v)$ be the total arrival rate at node j assuming that the external arrival rates are given by the vector v. Then

$$\lambda_j(v) = v_j + \sum_i \lambda_i(v) p_{ij}, \tag{2.4}$$

that is

$$\lambda(v) = (I - P^T)^{-1}v,$$

where P^T denotes the transpose of the matrix $P = ((p_{ij}))_{1\leq i,j\leq m}$. Assume that there exists $v \in \Gamma_\beta$ such that

$$\sum_i \alpha_{ki}\lambda_i(v) < 1, \quad k = 1, \ldots, k_0.$$

Then there is an allocation of effort that stabilizes the system. For example, select $u_i > \lambda_i(v)$ satisfying

$$\sum_i \alpha_{ki}u_i \leq 1, \quad k = 1, \ldots, k_0.$$

Recall that if we fix the arrival rates and the service allocations at v and u, then the stationary expectations become

$$E[Q_i] = \frac{\lambda_i(v)}{u_i - \lambda_i(v)}$$

2.1 Linear program for long-run average costs.

Let $c(q, u, v)$ be a nonnegative, lower semicontinuous function. The long-run average cost problem is to select U and V to minimize

$$J(U, V) = \limsup_{t\to\infty} E\left[\frac{1}{t}\int_0^t c(Q(s), U(s), V(s))ds\right].$$

Note that for bounded f, the fact that (1.1) is a martingale implies

$$\lim_{t\to\infty} E\left[\frac{1}{t}\int_0^t Af(Q(s), U(s), V(s))ds\right] = 0.$$

Define

$$\mu_t(C \times D) = E\left[\frac{1}{t}\int_0^t I_C(Q(s))I_D(U(s), V(s))ds\right].$$

If finiteness of $J(U, V)$ implies tightness of $\{\mu_t\}$, for example, if $c(q, u, v) = \sum_{i=1}^m q_i$ (recall that Γ^α and Γ_β are compact), then any limit point μ satisfies

$$\int_{\mathbb{Z}_0^m\times\Gamma^\alpha\times\Gamma_\beta} Af(q, u, v)\mu(dq \times du \times dv) = 0, \quad f \in \mathcal{D}(A), \tag{2.5}$$

and

$$\int_{\mathbb{Z}_0^m\times\Gamma^\alpha\times\Gamma_\beta} c(q, u, v)\mu(dq \times du \times dv) \leq J(U, V).$$

Consequently, we formulate the following linear programming problem: Minimize

$$\int_{\mathbb{Z}_0^m\times\Gamma^\alpha\times\Gamma_\beta} c(q, u, v)\mu(dq \times du \times dv) \tag{2.6}$$

subject to (2.5).

The relationship between controlled Markov processes and linear programming was recognized by Manne [9] in the context of finite Markov chains and developed by Stockbridge [11] for general Markov processes.

Theorem 2.1 *Suppose that for each $K > 0$,*

$$\{\mu : \int_{E\times F} c(q,u,v)\mu(dq \times du \times dv) \leq \bar{c}, \mu \text{ satisfies (2.5)}\} \tag{2.7}$$

is compact and is nonempty for $\bar{c}$ sufficiently large. Then there exists a solution of the linear programming problem.

It is not immediately obvious that the solution of the linear programming problem corresponds to the optimal solution of the stochastic control problem. The next result, which is a consequence of more general theorems in [6] or [2], ensures that it does.

Theorem 2.2 *Let $\mu \in \mathcal{P}(\mathbb{Z}_0^m \times \Gamma^\alpha \times \Gamma_\beta)$ satisfy (2.5), and suppose $\mu(dq \times du \times dv) = \mu_0(dq)\eta(q, du \times dv)$. Then there exists a stationary solution of the controlled martingale problem for A such that $U(s) = \int_{\Gamma^\alpha} u\eta(Q(s), du \times \Gamma_\beta)$, $V(s) = \int_{\Gamma_\beta} v\eta(Q(s), \Gamma^\alpha \times dv)$, and*

$$E[\int h(Q(s), U(s), V(s))] = \int h(q,u,v)\mu(dq \times du \times dv),$$

for every bounded, measurable h. In particular, the long-run average cost is given by (2.6).

Remark 2.3 *The general results in [11, 2, 6] require the use of relaxed controls. Relaxed controls are not needed here because the control enters the generator linearly.*

3 Heavy traffic limit.

There are two approaches to defining what is meant by a heavy traffic limit for the controlled system. One, which we will refer to as the *perturbed rate model*, is to fix nominal external arrival rates γ_i and then to require that the controlled external arrival and service rates be small perturbations of the nominal external arrival and total arrival rates. In terms of a scaling parameter n, we write

$$v_i = \gamma_i - \frac{1}{\sqrt{n}}v_i^0, \quad u_i = \lambda_i(\gamma) + \frac{1}{\sqrt{n}}u_i^0, \tag{3.8}$$

where

$$(u^0, v^0) \in F^n \equiv \{(u^0, v^0) : \lambda(\gamma) + \frac{1}{\sqrt{n}}u^0 \in \Gamma^\alpha, \gamma - \frac{1}{\sqrt{n}}v^0 \in \Gamma_\beta\}. \tag{3.9}$$

The magnitudes of v^0 and u^0 are constrained by penalizing the deviations from γ and $\lambda(\gamma)$. For example, take the running cost to be

$$c(q,u,v) = \frac{1}{\sqrt{n}}\sum_{i=1}^m q_i + \sum_{i=1}^m (|v_i^0| + |u_i^0|).$$

This approach is essentially the same as that taken in [10, 3].

The second approach, which we will refer to as the *constrained rate model,* is to assume that the constraints asymptotically determine unique nominal arrival rates. (See, for example, [8, 5, 4, 1, 12].) To formulate a version of this approach, we keep the p_{ij} and α_{ki} fixed and let the β_{li}^n vary with a normalizing parameter n. We assume that

β^n_{li} decreases to a limit β_{li} such that there is a unique $\gamma \in \Gamma_\beta$ satisfying $\lambda(\gamma) \in \Gamma^\alpha$. Uniqueness implies that for $i = 1, \ldots, m$, we must have

$$\sum_{\{k:\alpha_k \cdot \lambda(\gamma)=1\}} \alpha_k (I - P^T)^{-1} e_i > 0, \tag{3.10}$$

and

$$\gamma_i = 0 \quad \text{or} \quad \sum_{\{l:\beta_l \cdot \gamma = 1\}} \beta_{li} > 0. \tag{3.11}$$

In addition, assume that for some $b > 0$,

$$\beta_{li} \leq \beta^n_{li} \leq (1 + \frac{b}{\sqrt{n}})\beta_{li} \tag{3.12}$$

(in particular, $\beta_{li} = 0$ if and only if $\beta^n_{li} = 0$) and

$$\lim_{n\to\infty} \sqrt{n}(\beta^n_{li} - \beta_{li}) = \beta^0_{li}. \tag{3.13}$$

3.1 Asymptotic stability.

If we fix v^0 and u^0 in (3.8), then the stationary expectation satisfies

$$\frac{1}{\sqrt{n}} E[Q_i] = \frac{\lambda_i(\gamma - \frac{1}{\sqrt{n}} v^0)}{\sqrt{n}(\lambda_i(\gamma) + \frac{1}{\sqrt{n}} u^0_i - \lambda_i(\gamma - \frac{1}{\sqrt{n}} v^0)} \to \frac{\lambda_i(\gamma)}{u^0_i + \lambda_i(v^0)},$$

where $\lambda(v^0) = (I - P^T)^{-1} v^0$. Consequently, for the perturbed rate model, we have asymptotic stability if we can select $(u^0, v^0) \in F^n$ so that $u^0 + \lambda(v^0) > 0$.

Next, consider the constrained rate model. By the assumptions on β^n_{li}, $\beta^0_{li} \geq 0$. If $\beta^0 > 0$, then there exists a sequence $\gamma^n \in \Gamma_{\beta^n}$ such that $\gamma^n \to \gamma$ and

$$\sum_i \alpha_{ki} \lambda_i(\gamma^n) < 1, \quad k = 1, \ldots, k_0,$$

for each n. More generally, suppose that there exists $c > 0$ and $v^* \in [0, \infty)^m$ such that

$$\beta v^* \leq \beta^0 \gamma + c(\beta\gamma - \mathbf{1}), \qquad v^* \leq c\gamma, \tag{3.14}$$

and that there exists $\epsilon > 0$ such that

$$\alpha\lambda(v^*) \geq \epsilon\alpha\lambda(\gamma). \tag{3.15}$$

Without loss of generality, we can assume $\beta^0 v^* \leq c\mathbf{1}$. Note that if $\beta^0 > 0$, then the existence of v^* satisfying (3.14) and (3.15) is immediate. (Take $v^* = \epsilon\gamma$, for $\epsilon > 0$ sufficiently small.)

By (3.13), there exists $\rho_n < 1$ satisfying $\rho_n \to 1$ and

$$\sqrt{n}(\beta^n - \beta) \geq \rho_n \beta^0.$$

For $n > c^2$, define

$$\gamma^n = (1 + \frac{c}{n})\gamma - \frac{\rho_n}{\sqrt{n}} v^*.$$

Then

$$\begin{aligned}
\beta^n\gamma^n &= (\beta_n - \beta)((1+\frac{c}{n})\gamma - \frac{\rho_n}{\sqrt{n}}v^*) + (1+\frac{c}{n})\beta\gamma - \frac{\rho_n}{\sqrt{n}}\beta v^* \\
&\geq \frac{\rho_n}{\sqrt{n}}\beta^0((1+\frac{c}{n})\gamma - \frac{\rho_n}{\sqrt{n}}v^*) - \frac{\rho_n}{\sqrt{n}}\beta^0\gamma^T + (1+\frac{c}{n} - \frac{\rho_n}{\sqrt{n}}c)\beta\gamma + \frac{\rho_n}{\sqrt{n}}c\mathbf{1} \\
&= \frac{c\rho_n}{n^{3/2}}\beta^0\gamma - \frac{\rho_n^2}{n}\beta^0 v^* + (1+\frac{c}{n} - \frac{\rho_n}{\sqrt{n}}c)\beta\gamma + \frac{\rho_n}{\sqrt{n}}c\mathbf{1} \\
&\geq (1 - \frac{\rho_n}{\sqrt{n}}c)\beta\gamma + \frac{\rho_n}{\sqrt{n}}c\mathbf{1} \\
&\geq \mathbf{1},
\end{aligned}$$

so $\gamma^n \in \Gamma_{\beta^n}$. We have

$$\lambda(\gamma^n) = (I - P^T)^{-1}((1+\frac{c}{n})\gamma - \frac{\rho_n}{\sqrt{n}}v^*) = (1+\frac{c}{n})\lambda(\gamma) - \frac{\rho_n}{\sqrt{n}}\lambda(v^*),$$

and by (3.15), for n sufficiently large,

$$\frac{c}{n}\alpha\lambda(\gamma) < \frac{\rho_n}{\sqrt{n}}\alpha\lambda(v^*).$$

It follows that

$$\alpha\lambda(\gamma^n) \leq \mathbf{1} - \frac{\rho_n}{\sqrt{n}}\alpha\lambda(v^*) + \frac{c}{n}\lambda(\gamma) < \mathbf{1},$$

that is, the nth system is stable. In particular, we can take the service allocation to be of the form

$$u^n = \lambda(\gamma^n) + \frac{\delta^n}{\sqrt{n}}\mathbf{1}, \tag{3.16}$$

where

$$\delta^n = \min_k\{\rho_n\alpha_k \cdot \lambda(v^*) - \frac{c}{\sqrt{n}}\alpha_k \cdot \lambda(\gamma)\} \to \delta = \min_k \alpha_k \cdot \lambda(v^*).$$

Furthermore,

$$\lim_{n\to\infty} \sqrt{n}(\lambda(\gamma) - \lambda(\gamma^n)) = \lambda(v^*).$$

Consequently, defining

$$Z^n(t) = \frac{Q^n(nt)}{\sqrt{n}},$$

if we fix the external arrival rates as γ_n and the allocation policy as in (3.16), then the stationary means of Z^n satisfy

$$E[Z_i^n] = \frac{\lambda_i(\gamma^n)}{\delta^n} \to \frac{\lambda_i(\gamma)}{\delta}. \tag{3.17}$$

3.2 Limiting linear program.

We derive the limiting linear program for the perturbed rate model. The constrained rate model is more delicate and is a subject for future research.

Adding and subtracting partial derivatives where convenient, the generator for Z^n can be written

$$\begin{aligned}
&A^n f(z,u^0,v^0)\\
&\quad=\sum_{i=1}^{m}(\gamma_i-\frac{1}{\sqrt{n}}v_i^0)n(f(z+\frac{1}{\sqrt{n}}e_i)-f(z)-\frac{1}{\sqrt{n}}\partial_i f(z))\\
&\qquad+\sum_{i=1}^{m}(\lambda_i(\gamma)+\frac{1}{\sqrt{n}}u_i^0)p_{i0}n(f(z-\frac{1}{\sqrt{n}}e_i)-f(z)+\frac{1}{\sqrt{n}}\partial_i f(z))\\
&\qquad+\sum_{i,j=1}^{m}(\lambda_i(\gamma)-\frac{1}{\sqrt{n}}u_i^0p_{ij}n(f(z+\frac{1}{\sqrt{n}}(e_j-e_i))-f(z)-\frac{1}{\sqrt{n}}(\partial_j f(z)-\partial_i f(z)))\\
&\qquad-\sum_{i=1}^{m}v_i^0\partial_i f\\
&\qquad-\sum_{i=1}^{m}u_i^0(\partial_i f(z)-\sum_{j=1}^{m}p_{ij}\partial_j f(z))\\
&\qquad-\sum_{i}(\lambda_i(\gamma)+\frac{1}{\sqrt{n}}u_i^0)I_{\{z_i=0\}}p_{i0}n(f(z-\frac{1}{\sqrt{n}}e_i)-f(z))\\
&\qquad-\sum_{i,j}(\lambda_i(\gamma)+\frac{1}{\sqrt{n}}u_i^0)I_{\{z_i=0\}}p_{ij}n(f(z+\frac{1}{\sqrt{n}}(e_j-e_i))-f(z))\\
&\quad=L_0^n f(z,u^0,v^0)+(|u^0|+|v^0|)B_0 f(z,u^0/(|u^0|+|v^0|),v^0/(|u^0|+|v^0|))\\
&\qquad\qquad+\sum_{i=1}^{m}(\sqrt{n}\lambda_i(\gamma)+u_i^0)I_{\{z_i=0\}}C_{i,n}f(z),
\end{aligned}$$

where $|u^0|=\sum_{i=1}^m|u_i^0|$ and

$$B_0 f(z,u,v)=-(v+(I-P^T)u)\cdot\nabla f(z)$$

and

$$C_{i,n}f(z)=-p_{i0}\sqrt{n}(f(z-\frac{1}{\sqrt{n}}e_i)-f(z))-\sum_{j=1}^{m}p_{ij}\sqrt{n}(f(z+\frac{1}{\sqrt{n}}(e_j-e_i))-f(z)).$$

Note that we have used the fact that $\lambda(\gamma)=(I-P^T)^{-1}\gamma$.

We take $\mathcal{D}(A^n)$ to be $C_c^2(\mathbb{R}_0^m)$, the twice continuously differentiable functions on $\mathbb{R}_0^m=[0,\infty)^m$. Assuming that we can control the magnitude of u^0 and v^0, $L_0^n f(z,u^0,v^0)\to L_0 f(z)$ and $C_{i,n}f(z)\to C_i f(z)$, where

$$L_0 f(z)=\frac{1}{2}\sum_{1\le i,j\le m}a_{ij}^0\partial_i\partial_j f(z),$$

with

$$a_{ii}^0=\gamma_i+\lambda(\gamma)_i p_{i0}+\sum_{j\neq i}(\lambda(\gamma)_j p_{ji}+\lambda(\gamma)_i p_{ij}),\quad a_{ij}=-\lambda(\gamma)_i p_{ij}-\lambda(\gamma(_j p_{ji},\ i\neq j,$$

and

$$C_i f(z)=\partial_i f(z)-\sum_{j=1}^{m}p_{ij}\partial_j f(z).$$

Let the running cost function be

$$c(z,u^0,v^0)=\sum_{i=1}^{m}(z_i+|v_i^0|+|u_i^0|).$$

Noting that F^n defined in (3.9) is an increasing sequence of sets, we define F to be the closure of $\cup_n F^n$.

For each n, the optimal long-run average cost is given by the solution of the linear programming problem, minimize

$$\int_{[0,\infty)^m\times F^n} c(z,u^0,v^0)\mu(dz\times du^0\times dv^0)$$

among $\mu\in\mathcal{P}([0,\infty)^m\times F^n)$ satisfying

$$\int_{[0,\infty)^m\times F^n} A_n f(z,u^0,v^0)\mu(dz\times du^0\times dv^0)=0,\quad f\in\mathcal{D}(A_n). \tag{3.18}$$

Let μ^n be the minimizing probability measure. Define

$$\begin{aligned}
\mu_0^n(dz) &= \mu^n(dz\times F^n)\\
\mu_i^n(dz) &= \int_{F^n}(\sqrt{n}\lambda_i(\gamma)+u_i^0)I_{\{z_i=0\}}\mu^n(dz\times du^0\times dv^0)\\
\nu_0^n(dz\times H) &= \int_{F^n} I_H\Big(\frac{u^0}{|u^0|+|v^0|},\frac{v^0}{|u^0|+|v^0|}\Big)(|u^0|+|v^0|)\mu^n(dz\times du^0\times dv^0),
\end{aligned}$$

where μ_i^n is a measure on $D_i=\{z\in[0,\infty)^m : z_i=0\}$ and ν_0^n is a measure on $[0,\infty)^m\times\hat{F}^n$, $\hat{F}^n=\{(u/(|u|+|v|),v/(|u|+|v|)) : (u,v)\in F^n\}$. Since the optimal cost

$$\int_{[0,\infty)^m\times F^n} c(z,u^0,v^0)\mu^n(dz\times u^0\times v^0)=\int_{[0,\infty)^m}\sum_{i=1}^{m} z_i\mu_0^n(dz)+\nu_0^n([0,\infty)^m\times\hat{F}^n)$$

is bounded in n, it follows that $\{\mu_0^n\}$ is relatively compact in the weak topology and $\{\nu_0^n\}$ and $\{\mu_i^n\}$, $i=1,\ldots,m$ are relatively compact in the vague topology. If $(\mu_0,\nu_0,\mu_1,\ldots,\mu_m)$ is a limit point of $(\mu_0^n,\nu_0^n,\mu_1^n,\ldots,\mu_m^n)$, for each $f\in C_c^2([0,\infty)^m)$,

$$\int_{[0,\infty)^m} L_0 f(z)\mu_0(dz)+\int_{[0,\infty)^m\times\hat{F}} B_0 f(z,u,v)\nu_0(dz\times du\times dv)+\sum_{i=1}^{m}\int_{[0,\infty)^m} C_i f(z)\mu_i(dz)=0,$$

and

$$\liminf_{n\to\infty}\int_{[0,\infty)^m\times F^n} c(z,u^0,v^0)\mu^n(dz\times du^0\times dv^0)\geq\int_{[0,\infty)^m}\sum_{i=1}^{m} z_i\mu_0(dz)+\nu_0([0,\infty)^m\times\hat{F}). \tag{3.19}$$

Consequently, the left side of (3.19) is bounded below by the solution of the linear programming problem, minimize

$$\int_{[0,\infty)^m}\sum_{i=1}^{m} z_i\mu_0(dz)+\nu_0([0,\infty)^m\times\hat{F})$$

subject to

$$\int_{[0,\infty)^m} Lf(z)\mu_0(dz)+\int_{[0,\infty)^m\times F} B_0 f(z,u,v)\nu_0(dz\times du\times dv)+\sum_{i=1}^{m}\int_{[0,\infty)^m} C_i f(z)\mu_i(dz)=0. \tag{3.20}$$

The following theorem is a special case of results in [7].

Theorem 3.1 *Suppose μ_0, ν_0, and $\mu_1, \ldots, \mu_m$ satisfy (3.20). Then there exist $(Z, \Lambda_0, \Lambda_i)$ such that*

$$f(Z(t)) - \int_0^t Lf(Z(s))ds - \int_{[0,\infty)^m \times F \times [0,t]} B_0 f(z,u,v)\Lambda_0(dz \times du \times dv \times ds) - \sum_{i-1}^{m} \int_{D_i \times [0,t]} C_i f(z)\Lambda_i(dz \times ds)$$

is a martingale for each $f \in C_c^2$, and Z and the increments of $\Lambda_0, \ldots, \Lambda_m$ are stationary with

$$\begin{aligned} E[f(Z(t))] &= \int f d\mu_0 \\ E[\Lambda_0(G \times H \times [0,t])] &= \nu_0(G \times H)t \\ E[\Lambda_i(G \times [0,t])] &= \mu_i(G)t \end{aligned}$$

References

[1] Williams, R. J. and Bell, S. L. Dynamic scheduling of a system with two parallel servers in heavy traffic with complete resource pooling: Asymptotic optimality of a continuous review threshold policy. *Preprint* (1999).

[2] Bhatt, A. G. and Borkar, V. S. Occupation measures for controlled Markov processes: Characterization and optimality. *Ann. Probab.* 24 (1996), 1531-1562.

[3] Kushner, Harold J. and Chen, Y. N. Optimal control of assignment of jobs to processors under heavy traffic. *Preprint* (1999).

[4] Harrison, J. Michael. Heavy traffic analysis of a system with parallel servers: symptotic optimality of discrete-review policies. *Ann. Appl. Probab.* 8 (1998), 822-848.

[5] Kelly, F. P. and Laws, C. N. Dynamic routing in open queueing networks: Brownian models, cut constraints and resource pooling. *Queueing Systems* 13 (1993), 47-86.

[6] Kurtz, Thomas G. and Stockbridge, Richard H. Existence of Markov controls and characterization of optimal Markov controls. *SIAM J. Cont. Optim.* 36 (1998), 609-653.

[7] Kurtz, Thomas G. and Stockbridge, Richard H. (in preparation)

[8] Laws, C. N. Resource pooling in queueing networks with dynamic routing. *Adv. Appl. Prob.* 24 (1992), 699-726.

[9] Manne, A. S. Linear programming and sequential decisions, *Management Sci.*, **6** (1960), 259-267.

[10] Martins, Luiz Felipe and Kushner, Harold J. Routing and singular control for queueing networks in heavy traffic. *SIAM J. Control and Optimization* 28 (1990), 1209-1233.

[11] Stockbridge, Richard H. Time-average control of martingale problems: A linear programming formulation, *Ann. Probab.*, **18** (1990), pp. 206–217.

[12] Williams, R. J. On dynamic scheduling of a parallel server system with complete resource pooling. *Preprint* (1999).

Stationary Reflected Lévy Processes in Stochastic Networks*

TAKIS KONSTANTOPOULOS
Department of Electrical
and Computer Engineering
University of Texas at Austin
Austin, TX 78712, USA
`takis@ece.utexas.edu`

GÜNTER LAST
Institut für Mathematische Stochastik
Technische Universität Braunschweig
Pockelsstraße 14, Postfach 33 29
38023 Braunschweig, Germany
`g.last@tu-bs.de`

Abstract

Motivated by recent approximation results for traffic processes arising in high-speed network applications with highly variable durations, we consider Lévy processes of spectrally positive type as inputs to networks and derive a number of results regarding the performance of such systems. The results include a representation formula for the Skorokhod reflection mapping in dimension 1, Little's laws, tail asymptotics for steady state probabilities and the question of existence product form of the stationary distribution. The paper is supposed to be a summary, rather than a complete account, of recent work.

1 Introduction

In this paper we summarize a few recent results on the use of Lévy processes for modeling inputs in stochastic networks. We view these networks as models of communication networks with highly variable, possibly long-range dependent traffic. In fact, we maintain that Lévy processes of "spectrally positive type" are useful in obtaining macroscopic estimates and approximations for networks.

The type of networks we examine frequently arise as functional approximations of more "microscopic models". The meaning is clear if one recalls the existing body of work in modeling of manufacturing systems using Brownian motions. The often non-central limit type of approximations for stochastic models "possessing high variability" lead to processes with independent increments but discontinuous paths. In fact, the types of discontinuities are quite "wild". A need to understand the dynamics of these "extended fluid queueing models" leads us to a derivation of integral representations of Skorokhod reflections, Little's laws, (extended) Pollaczeck-Khinchine and Takács-like formulae, etc. We make an effort to present these things below, along with a unifying tool, that of stochastic calculus.

The article is organized as follows: First, we define the types of Lévy processes we consider. Then we look at the concept of a generalized fluid queue. We present various results on its dynamics and performance. Subsequently, we define a network of single-class type and look at things we can say about its steady-state; in particular, about

---

*Research supported in part by NSF Grants NCR-95-02582 and ANI-99-03495

the tail behavior of its steady state distribution, and the possibility of product form solutions. We refer a[n] (unfortunately incomplete) discussion of the literature to the end of the article. For more details on the bibliography, as well as the proofs of the various statements in this paper, we refer to [8] and [9].

2 Spectrally positive Lévy processes in $\mathbb{R}^d$

A general Lévy process $\{X_t\}$ in $\mathbb{R}^d$ (with t running over $[0, \infty)$ or $\mathbb{R}$) is characterized by its Lévy-Khinchine formula

$$Ee^{i\theta^\top X_t} = \exp\left\{ t \left[ia^\top\theta - \frac{1}{2}\theta^\top\sigma\sigma^\top\theta + \int_{\mathbb{R}^d} (e^{i\theta^\top x} - 1 - i\theta^\top x \mathbf{1}\{|x| \le 1\})\nu(dx) \right] \right\}, \quad \theta \in \mathbb{R}.$$

where ν is a sigma-finite measure on $\mathbb{R}^d - \{0\}$, integrating $|x|^2 \wedge 1$. The vector a corresponds to a linear drift, while σ is a positive-definite matrix corresponding to a Brownian motion. The last term corresponds to the purely discontinuous part of the Lévy process. A good reference for these concepts is Bertoin [2]. We say that X is *spectrally positive*[1] iff ν is concentrated on the positive orthant. The terminology is ours–and it is a natural extension of its one-dimensional counterpart. Spectral positivity implies existence of the Laplace exponent

$$\Phi(\theta) = -\log Ee^{-\theta^\top X_1} = a^\top\theta - \frac{1}{2}\theta^\top\sigma\sigma^\top\theta + \int_{\mathbb{R}^d_+} (1 - e^{-\theta^\top x} - \mathbf{1}\{|x| \le 1\}\theta^\top x)\nu(dx), \quad \theta \in \mathbb{R}^d_+.$$

The pure jump part of such a process (call it X again for simplicity) is α-self-similar iff for any $\kappa > 0$, $\{X_{\kappa t}\}$ is identical in law to $\kappa^{1/\alpha}\{X_t\}$. The constant α is necessarily in the interval $(0, 2)$. When $1 < \alpha < 2$, then then the Lévy measure is of special form, best described in polar coordinates:

$$\nu(dr, dz) = \frac{dr}{r^{\alpha+1}}\tau(dz), \quad r > 0, \quad z \in S^{d-1}_+, \tag{1}$$

where $S^{d-1}_+ = \{x \in \mathbb{R}^d_+ : |x| = 1\}$, and τ is a finite Borel measure on S^{d-1}_+. By manipulating the Laplace exponent (say $\sigma = 0$) we can integrate the Lévy part and arrive at

$$\Phi(\theta) = \mu^\top\theta - \gamma \int_{S^{d-1}_+} (\theta^\top z)^\alpha \tau(dz). \tag{2}$$

Thus, such a process is a (continuous) convex linear combination of one-dimensional spectrally positive α-self-similar Lévy processes. The coefficient γ is just a constant coming from the integration and can be absorbed into τ. The coefficient μ is the mean of the process:

$$EX_t = t\mu.$$

When $\alpha \le 1$, the mean of X does not exist. The case $\alpha < 1$ corresponds to a multidimensional self-similar subordinator, i.e. a process which is a convex linear combination of non-negative (and thus increasing) self-similar one-dimensional Lévy processes.

---

[1] More precisely, it has a *spectrally positive jump part*

3 Stochastic fluid systems

Using nothing more than the stationarity of the increments of a Lévy process X we define a stationary stochastic fluid queue as a one-dimensional process Z that is (i) stationary, and (ii) solves the one-dimensional standard Skorokhod reflection problem, viz.,

$$Z_t = \sup_{s<u\le t} (X_t - X_u) \vee (Z_s + X_t - X_s), \quad -\infty < s < t < \infty. \tag{3}$$

Note here that if X is originally defined only on $[0, \infty)$ it can be extended to $\mathbb{R}$ due to the stationarity of its increments; it is assumed that this extension has been done. Also, X_0 can be taken to be 0 (no harm done: the above formula depends only on the increments of X). If X_1 has finite first moment μ then conditions for the existence and uniqueness of Z are well-known: $\mu < 0$. More generally, the problem is a bit more subtle. If, however,
(C1) $\inf_{s\le 0} X_s > -\infty$, a.s., and
(C2) $\liminf_{t\to\infty} X_t = -\infty$, a.s.,
then there is a unique stationary solution Z.

Such a generalized queue can be seen to have many similarities with a classical queue. Let us assume that X has finite variation[2] Then X can be decomposed as the difference of two increasing processes[3]

Theorem 1. *Suppose that Z is defined as a stationary solution to (3), with X being a finite variation process with stationary increments. Let $X = A - B$ be any decomposition of X as difference of two increasing processes. Then*

$$Z_t = \int_{-\infty}^{t} \mathbf{1}(Z_u > B_t - B_u)\, dA_u^c + \int_{(-\infty,t]} \frac{(Z_u - B_t + B_u])^+ \wedge \Delta A_u}{\Delta A_u}\, dA_u^d, \tag{4}$$

where A^c, respectively A^d, is the continuous, respectively discrete, part of A, and $\Delta A_u := A_u - A_{u-}$.

In fact, we claim that (4) is completely equivalent to (3). It is interesting to note that the right hand side of (4) depends only on $A - B$ and not on the individual terms A, B, but this is not obvious from the above expression. The formula also allows the possibility that A, B have common jumps.

A nice consequence of (4) is the so-called Little's law. Thinking of A as a random measure, and assuming it has finite rate α, let P_A be the Palm probability of P with respect to A. Similar notation is used for other Palm probabilities.

Theorem 2. *With the assumptions of Theorem 1, and if Z has finite first moment, then*

$$EZ_0 = \alpha^c E_{A^c} B_-^{\leftarrow}(Z_{0-}) + \alpha^d E_{A^d} \int_0^\infty \frac{(Z_{0-} + \Delta A_0 - B_u)^+ \wedge \Delta A_u}{\Delta A_u}\, du,$$

where α^c, α^d are the rates of A^c, A^d, respectively, E_A, E_{A^c}, E_{A^d} are expectations with respect to the Palm probabilities P_A, P_{A^c}, P_{A^d}, and $B_-^{\leftarrow}$ is the left-continuous version of the generalized inverse of B, i.e. $B_-^{\leftarrow}(x) := \inf\{t \ge 0 : \ B_t \ge x\}$.

---

[2] i.e., bounded variation on finite intervals; this excludes α-self-similar Lévy processes with $\alpha > 1$, but includes many interesting subordinators, such as a Gamma process.

[3] All processes are taken to be right continuous.

It is worth looking at a couple of special cases:

1. Suppose that A is continuous and B deterministic, say $B_t = t$. then

$$EZ_0 = \alpha E_A Z_0.$$

This is a classical fluid queue with linear server.

2. Suppose that A is general, and $B_t = t$. Then

$$EZ_0 = \alpha E_A Z_{0-} + \frac{1}{2}\alpha^d E_{A^d} \Delta A_0.$$

This is a generalized fluid queue with linear server.

Actually, the case of linear server allows a generalized version of Little's law.

Theorem 3. *Under the assumptions of Theorem 2, and if, in addition, $B_t = t$, then for any measurable function $\psi : \mathbb{R}_+ \to \mathbb{R}$,*

$$E\psi(Z_0) = (1-\alpha)\psi(0) + \alpha E_A \frac{\int_{Z_{0-}}^{Z_0} \psi(x)\, dx}{Z_0 - Z_{0-}},$$

with the understanding that the fraction in the right hand side equals $\psi(Z_0)$ on the event $\{Z_0 = Z_{0-}\}$.

Here is an application of the formulas in a classical situation: consider a Gamma(a, b) process A. It is a Lévy process with increasing sample paths and L'evy measure $\nu(dx) = a^{-1}e^{-bx}dx$, for $x > 0$. It is also referred to as the Moran gamma process, in connection to the theory of dam processes. (See Kingman [7], ch. 9, for a superb treatment of this process.) It can be seen that $Ee^{-\theta A_1} = a\log(1+\theta/b)$, $\theta > 0$. Assume that $a < b$. Then $EA_1 = a/b < 1$, so if the server is $B_t = t$, then the system is stable. The steady-state distribution can be found by choosing ψ above to be an exponential. Thus,

$$Ee^{-\theta Z_0} = \frac{1 - \frac{a}{b}}{1 - \frac{a}{\theta}\log(1+\theta/b)}.$$

4 Spectral positivity in dimension one

The more interesting processes for us are spectrally positive α-self-similar Lévy processes with $1 < \alpha < 2$. Although such processes have finite first moment, they have infinite variation sample paths and thus do not fall in the category of processes described in the previous section. To analyze Z with an X of this type is a classical problem. However, to find its distribution using Itô's formula directly appears to be new and the quickest method around. Assume that the Laplace exponent of X is

$$\Phi(\theta) = \mu\theta - \frac{1}{2} + \int_0^\infty (1 - e^{\theta x} - \theta x)\nu(dx),$$

where ν integrates x away from the origin (so that $EX_1 = \mu < \infty$), and where we have also added a Brownian motion for the sake of generality. Let f be a suitable smooth function and write

$$f(Z_1) - f(Z_0) = \int_0^1 f'(Z_{u-})dZ_u + \frac{1}{2}\int_0^1 f''(Z_{u-})d[Z]^c_u + \sum_0^1 [\Delta f(Z_u) - f'(Z_{u-})\Delta Z_u],$$

where $\Delta f(Z_u) := f(Z_u) - f(Z_{u-})$, and $\Delta Z_u := Z_u - Z_{u-}$. (See, e.g., Kallenberg [4], p. 439.) Substituting for the individual terms and taking expectations on both sides, we arrive at

$$\mu E f'(Z_0) + \frac{\sigma^2}{2} E f''(Z_0) + E \int_0^\infty [f(Z_0 + x) - f(Z_0) - f'(Z_0)x]\, \nu(dx) = 0.$$

Next, choosing f to be an exponential yields a complete solution for the Laplace transform of Z_0:

$$E e^{-\theta Z_0} = \frac{\mu\theta}{\Phi(\theta)}.$$

A particularly important case is the following: Let ν be of the form (1). Since $d = 1$, the radial part is just a constant, say c: $\nu(dx) = c x^{-\alpha-1} dx$, $x > 0$, $1 < \alpha < 2$. It can be readily seen that

$$\Phi(\theta) = \mu\theta - c'\theta^\alpha,$$

where $c' = c\Gamma(2-\alpha)(\alpha-1)^{-1}\alpha^{-1}$. The Laplace transform of Z_0 is

$$E e^{-\theta Z_0} = \frac{1}{1 + |c'\mu^{-1}|\theta^{\alpha-1}}.$$

The limiting case $\alpha \to 2$ yields (formally) an exponential distribution (which is the distribution of the ultimate maximum of a Brownian motion with negative drift). The case $1 < \alpha < 2$ corresponds to a so-called Mittag-Leffler distribution, which is a "fractional" generalization of the exponential. We cannot invert this Laplace transform explicitly, but we can approximate it by power series. The behavior of the tail of the distribution of Z_0 is known: it is a regularly-varying function of index $-\alpha$.

5 Lévy networks

We return to the question of defining and analyzing simple networks with Lévy inputs. We first define the system. We are given a $d \times d$ reflection matrix R such that $I - R$ is non-negative, with spectral radius strictly smaller than 1. Let X be a process with stationary increments, and values in $\mathbb{R}^d$. The right way to think about the network process Z is by reflecting X in the usual Skorokhod manner:

$$Z_t - Z_s = X_t - X_s + R(L_t - L_s), \tag{5}$$

where L is an increasing process, such that

$$\sum_{i=1}^{d} \int 1(Z_s^i > 0) dL_s^i = 0.$$

This is a generalization of (3). To start with, one asks a stability question: when is it true that (5) has a unique stationary solution? The existence question can be directly deduced from a paper by Kella and Whitt [5]. In particular,

Theorem 4. *Suppose that X is a Lévy process in $\mathbb{R}^d$ with finite mean. Let*

$$EX_t = \mu t,$$

such that

$$R^{-1}\mu < 0.$$

Then there exists a stationary process Z satisfying (5).

Allow now the process X to be a Lévy process of type (2), and add a Brownian motion for the sake of generality:

$$\Phi(\theta) := -\log E e^{-\theta^{\mathsf{T}} X_1} = \mu^{\mathsf{T}}\theta - \frac{1}{2}\theta^{\mathsf{T}}\sigma\sigma^{\mathsf{T}}\theta - \int (\theta^{\mathsf{T}} z)^{\alpha}\tau(dz). \tag{6}$$

Let also Z be a stationary process satisfying (5). The question is what can one say about the distribution of Z_0. Let

$$\varphi(\theta) := E e^{-\theta^{\mathsf{T}} Z_0}.$$

It can be seen that spectral positivity implies that the process L has continuous paths. Define the "expected time spent on the boundary $\{x_j = 0\}$" of the positive orthant, by

$$\beta_j(A) := \frac{1}{t} E \int_0^t \mathbf{1}\{Z_{s-} \in A\} dL_s^j.$$

$j = 1, \dots, d$. The following theorem expresses the fact that knowledge of these "boundary measures" implies knowledge of the distribution of Z_0.

Theorem 5. *Under the foregoing assumptions and notations,*

$$\varphi(\theta) = \frac{-\sum_j \hat{\beta}_j(\theta)\theta^{\mathsf{T}} r_j}{\Phi(\theta)},$$

where Φ is as in (6)*, r_j is the j-th column of R, and $\hat{\beta}_j(\theta) := \int_{\mathbb{R}_+^d} e^{-\theta^{\mathsf{T}} z}\beta_j(dz)$.*

The theorem is proved by an extension of the method explained in dimension 1. A consequence of the above theorem is:

Corollary 1. *If either $(\sigma\sigma^T)_{ii} > 0$ or $\int z_i \tau(dz) > 0$, then $\lim_{\theta_i \to \infty} \varphi_i(\theta_i) = 0$ and $P(Z_0^i = 0) = 0$.*

This is very different from the subordinator or classical queueing case: there the stationary distribution for Z^i takes value 0 with positive probability. The absence of atom at 0 here is rather important.

A nice consequence of the above is that we can obtain tail asymptotics for Z_0. To this end, we have a lemma with a "balance relation" involving P_j, the Palm probability of P with respect to the stationary random measure defined by the increasing process L^j. Let λ_j be the rate of L^j.

Lemma 1. *For any $u > 0$,*

$$\int e^{-ux} P(\theta^{\mathsf{T}} Z_0 \geq x) dx = \frac{\Phi(u\theta) - u\theta^{\mathsf{T}}\mu}{u\Phi(u\theta)} - \frac{u}{\Phi(u\theta)} \sum_{j=1}^{d} \lambda_j \theta^{\mathsf{T}} r_j \int e^{-ux} P_j(\theta^{\mathsf{T}} Z_0 \geq x) dx.$$

The lemma is proved using the relation of φ with respect to the boundary measures. Finally, we have:

Theorem 6. *Assume that* $d = 2$ *and let* X *be a Lévy process defined by its Laplace exponent* $\Phi(\theta)$ *as in* (6) *with* $\sigma \equiv 0$. *Let* Z *be a stationary solution to the Skorokhod reflection problem with reflection matrix* R. *Then there exist constants* $c_1, c_2 > 0$, *depending only on the parameters* α, μ, R, τ *of the model, such that*

$$c_1 \le \liminf_{x \to \infty} x^{\alpha-1} P(|Z_0| \ge x) \le \limsup_{x \to \infty} x^{\alpha-1} P(|Z_0| \ge x) \le c_2.$$

The theorem is proved on the basis of the previous lemma, using a Tauberian theorem and some algebra. It should be clear that the result is true for any d. However, its proof may be complicated.

A classical question in the performance analysis of standard queueing network models is the existence of a set of conditions under which the stationary distribution has independent marginals ("product form" distribution). It is known (see Harrison and Williams [3]) that the answer is affirmative when $\nu \equiv$, i.e. when the input is purely Brownian. However, in the spectrally positive évy case we obtain:

Theorem 7. *Let* $c = \sigma\sigma^\top$. *Assume for all* $i = 1, \dots, d$ *that either* $c_{ii} > 0$ *or* $\int z_i \tau(dz) > 0$ *and that the distribution of* Z_0 *has product form. Then*

$$c_{ij} = \frac{r_{ij} c_{jj}}{2 r_{jj}} + \frac{r_{ji} c_{ii}}{2 r_{ii}} \quad i, j = 1, \dots, d,$$

and

$$\tau = \sum_{i=1}^{d} \kappa_i \delta_{e_i},$$

where e_i *is the unit vector in the* i*-th direction and* δ_{e_i} *is the Dirac measure at* e_i, *whereas the* κ_i *are constants. Moreover, if* $\int z_i \tau(dz) > 0$, *then* $r_{ij} = 0$ *for all* $j \ne i$.

The meaning of this theorem is that in the presence of a non-trivial spectrally positive component, the stationary distribution cannot be of product form unless the network is trivial, i.e. all inputs independent (the marginals of X independent) and all nodes non-interacting (R diagonal).

6 Discussion

Here is a brief motivation for the type of models considered in this paper. The discussion is taken from Konstantopoulos and Lin (1998). Consider a simple but reasonable model for high-speed communication network traffic defined as follows. We are given a marked point process on the real line with points T_n and corresponding marks ξ_n, $n \in \mathbb{Z}$. In the simplest case, T_n form a Poisson (λ) process, while $\{\xi_n\}$ are i.i.d. (positive) random measures, independent of $\{T_n\}$. We define an increasing process $\{S_t, t \ge 0\}$ by

$$S_t := \sum_{n \in \mathbb{Z}} \xi_n((0, t] - T_n), \quad t \ge 0.$$

As an example, consider a deterministic increasing function $\{\zeta(t), t \geq 0\}$ and an i.i.d. sequence of positive random variables $\{\tau_n\}$ and let ξ_n be defined by

$$\xi_n(B) := \int_B d\zeta(t \wedge \tau_n)$$

for B a Borel subset of $\mathbb{R}_+$. The interpretation is as follows: a source arrives at communications facility at time T_n and transmits for a duration equal to τ_n according to the function ζ. The superposition of these sources is the arrival process. Thus, S_t equals the total volume of data transmitted by all sources from time 0 up to time t.

The kind of random variables that give rise to interesting models are those that have heavy tails, for they are supposed to be responsible for the well-recognized long-range dependence phenomenon in traffic data; see Konstantopoulos and Lin [10] and references therein. Thus, we assume that τ_n have common distribution $F(t) = P(\tau_n \leq t)$, such that the function $t \mapsto 1 - F(t)$ has regularly varying tail with index $-\alpha$ (see, e.g., Bingham et al. [1]), where $1 < \alpha < 2$. It is easy to observe that S is a long-range dependent process, at least in the sense that its covariance function is non-integrable.

In a practical situation, the superposition of sources acts as an arrival process in a buffer whose contents are transmitted at a constant rate, larger than the total mean input rate. Doing queueing theory with an arrival process like S (also known as an "$M/G/\infty$" input) is a way of analyzing the stochastic contents of a buffer (see, e.g., Likhanov et al. [11]). One can quantitatively obtain the "queueing results" by considering asymptotics for a properly scaled version of S. It turns out that the correct scaling is of the form

$$\tilde{S}_t^n := c(n)^{-1}(S_{nt} - ES_{nt}), \quad t \geq 0, \quad n = 1, 2, \ldots,$$

where $c(n)$ has regularly varying tail.

For example, if $\zeta(t) = t$, meaning that sources transmit at constant rates, then $c(n)$ has regularly varying tail of index $1/\alpha$. The limit of the sequence $\{\tilde{S}^n\}_n$, as $n \to \infty$, is (at least as concerns the finite dimensional distributions) a zero-mean, spectrally positive Lévy process (with no Brownian part) and (consequently) Lévy measure of the form

$$\nu(dx) = cx^{-\alpha-1}dx, \quad x > 0,$$

which is the same as the case discussed earlier. The "queueuing theoretic" analysis of the buffer contents explained above is equivalent to studying the reflection of the process S, or, approximately, the reflection of its approximating Lévy process. This yields a stationary distribution of Mittag-Leffler type. For related work justifying the use of Lévy processes see also Resnick and van den Berg [12], and Whitt [13, 14]).

Continuing with these applications in mind, it appears that self-similar spectrally positive Lévy processes are important in the macroscopic modeling of traffic data and, consequently, in the performance analysis of stochastic models of high-speed networks. Thus, what we did in the last three sections is a queueing networks generalization of what was described above.

Independently of these applications, the question of product-form in Lévy networks[4] was asked by Kella in his 1998 paper [6] (also look therein for earlier work of the author and co-workers on similar questions) but for different types of Lévy processes.

---

[4] if one can coin in such a terminology–in analogy to that of more standard "Brownian networks"

References

[1] Bingham, N.H, Goldie, C.M. and Teugels, J.L. (1987) *Regular Variation.* Encyclopedia of Mathematics and its Applications, Vol. **27**, Cambridge University Press.

[2] Bertoin, J. (1996) *Lévy processes.* Cambridge University Press, Cambridge.

[3] Harrison, J.M. and Williams, R.J. (1987) Brownian models of open queueing networks with homogeneous customer populations. *Stochastics* **22**, 77-115.

[4] Kallenberg, O. (1997) *Foundations of Modern Probability.* Springer, New York.

[5] Kella, O. and Whitt, J. (1996) Stability and structural properties of stochastic storage networks. *J. Appl. Prob.* **33**, 1169-1180

[6] Kella, O. (1998) Non-product form of two-dimensional fluid networks with dependent Lévy inputs. Preprint, Dept. of Statistics, The Hebrew University of Jerusalem.

[7] Kingman, J.F.C. (1993) *Poisson Processes.* Oxford University Press, Oxford.

[8] Konstantopoulos, T. and Last, G. (1999) On the dynamics and performance of stochastic fluid systems. Preprint; available at: `http://www.math.tu-bs.de/stochastik/last.htm`

[9] Konstantopoulos, T. and Last, G. (1999) On stationary reflected Lévy processes. Preprint; available at: `http://www.math.tu-bs.de/stochastik/last.htm`

[10] Konstantopoulos, T. and Lin, S.J. (1998) Macroscopic models for long-range dependent network traffic. *Queueing Systems* **28**, 215-243.

[11] Likhanov, N., Tsybakov B. and Georganas, N.D. (1995) Analysis of an ATM buffer with self-similar ("fractal") input traffic, *Proc. IEEE INFOCOM'95*, Boston.

[12] Resnick, S. and van den Berg, E. (1999) Weak convergence of high-speed network traffic models. School of OR&IE, Cornell University, Technical Report No. 1236.

[13] Whitt, W. (1998) Non-Brownian FCLTs for the single-server queue. Preprint, AT&T Labs.

[14] Whitt, W. (1999) Limits for cumulative input processes to queues. Preprint, AT&T Labs.

On the impact of variability on the buffer dynamics in IP networks *

Youngmi Joo, Vinay Ribeiro,
Anja Feldmann, Anna C. Gilbert, and Walter Willinger

Abstract

The main objective of this paper is to demonstrate in the context of a simple TCP/IP-based network that depending on the underlying assumptions about the inherent nature of the variability of network traffic, very different conclusions can be derived for a number of well-studied and apparently well-understood problems in the areas of traffic engineering and management. For example, by either fully ignoring or explicitly accounting for the empirically observed variability of network traffic at the source level, we provide detailed *ns-2*-based simulation results for two commonly-used traffic workload scenarios that can give rise to fundamentally different buffer dynamics in IP routers. We also discuss a set of *ns-2* simulation experiments to illustrate that the queueing dynamics within IP routers can be qualitatively very different depending on whether the observed variability of measured network traffic over small time scales is assumed to be in part endogenous in nature (i.e., due to TCP's feedback flow control mechanism, which is "closed loop") or is exogenously determined, resulting in an "open loop" characterization of network traffic arriving at the routers.

1 Introduction

Traffic characterization and modeling are generally viewed as important first steps toward understanding and solving network performance-related problems. At the same time, there is little disagreement that the resulting understanding of and solutions to network performance problems are only as good and complete as the underlying assumptions on the usage of the network and the nature of the traffic that it carries. The main goal of this paper is to highlight with a toy example of a TCP/IP network the extent to which assumptions underlying the nature of network traffic can influence practical engineering decisions. More specifically, using the *ns-2* network simulator [1], we illustrate how by either implicitly accounting for or explicitly ignoring the empirically observed variability of network traffic over large and small time scales, a range of different and at times opposing conclusions can be drawn about the inferred buffer dynamics for IP routers. While there are many known causes for the observed variability in measured TCP traffic (e.g., see [4]), in this paper we focus on just two of them. On the one hand, we consider a known cause for variability of the packet rate process over large time scales; that is, high variability at the connection level resulting in self-similar scaling. On the other hand, we investigate a suspected cause for variability of the rate process over small time

---

[*] Y. Joo is with the Department of Electrical Engineering, Stanford University, CA 94305-9030, email: jym@leland.stanford.edu. V. Ribeiro is with the ECE Department, Rice University, Houston, TX 77005, email: vinay@rice.edu. A. Feldmann, A. Gilbert, and W. Willinger are with AT&T Labs-Research, Florham Park, NJ 07932-0971, email: {anja, agilbert, walter}@research.att.com.

scales; that is, the dynamics or actions of TCP resulting in complex scaling behavior of the traffic over small time scale that is consistent with multifractal scaling [5].

As far as large time scale variability is concerned, the extreme case of no variability can be achieved by requiring at the application layer that each active source has to transfer an essentially infinite file for the duration of the entire simulation. This *infinite source model* lacks the natural variability that has been observed in measured data at the application level and, in turn, lacks any non-trivial large time scale variability in the corresponding rate process [11, 3, 6]. In contrast, large time scale variability that is consistent with self-similar scaling over those time scales and has become a trademark of measured traffic rate processes can be achieved in a parsimonious manner by explicitly accounting for an adequate level of variability at the application layer [15, 14]. In fact, by replacing the infinite source model by a SURGE-like workload generator [2], that is, by changing an infinite file transfer into an application that imitates a typical Web session (with appropriately chosen heavy-tailed distributions to match the observed variability of various Web-related items such as session length and size, size of requested Web pages), we obtain a *Web-user source model* that automatically generates the desired large time scale variability in the aggregate rate process. These workload models are described in more details in Section 2.

Subsequently, in a first set of *ns-2* simulation experiments, we demonstrate in Section 3 how the two source models lead to two qualitatively very different queueing dynamics in the router. While the queueing behavior with the infinite source models is completely dominated by synchronization effects, these effects essentially disappear when the sources are Web users. Using the terminology originally due to V. Jacobson, we illustrate how the presence of many short-lived TCP connections or "mice" – a salient feature of our Web workload model – completely changes the buffer dynamics that has been noted in the past when the network load is entirely due to the long-lived TCP connections or "elephants" of the infinite source model. For earlier studies on TCP dynamics assuming infinite source models, see for example [9, 12, 16, 17]).

In a second set of *ns-2* simulation experiments, we focus in Section 4 on the small time scale variability of the packet rate process and its likely cause, that is, the feedback flow control mechanism that is part of the TCP/IP protocol suite [5, 4]. In particular, we summarize the results of our studies on how the network through the TCP end-to-end congestion control algorithm shapes (with some delay) the packet flow emitted from the different sources, which in turn alters the rate process that arrives at the IP router for buffering (which in turn impacts the level of congestion, etc.). The results are obtained by performing a number of related *closed loop* and *open loop* simulations and comparing them on the basis of some commonly-used performance criteria. Here, by "closed loop" we mean a *ns-2* simulation with a fixed networking configuration, including buffer size in the router(s), link bandwidths, delays, etc., and where all hosts use TCP. In contrast, "open loop" means we collect a packet trace from a particular *ns-2* simulation run (or alternatively, from a link within the Internet) and use it to perform a trace-driven simulation of a queueing system that represents our IP router. Note that trace-driven simulations cannot account for the capabilities of the network to shape and thus alter the offered traffic to the queue.

We conclude in Section 5 by commenting on a number of admittedly unrealistic assumptions regarding our simulation configuration. However, despite these over-simplifications and a number of other shortcomings of our studies, we believe that the findings reported in this paper will increase the overall awareness that it is in general easy to draw conclusions based on infinite source models and/or open loop systems and their performance, but that the real challenges lie in convincingly demonstrating that these conclusion either still hold or become invalid for realistically loaded networks and/or the corresponding closed loop systems and their performance.

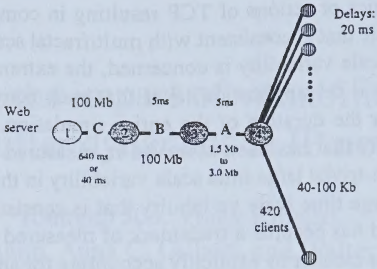


Figure 1: Network configuration

2 The Simulation Setup

In this section, we give a description of the networking configuration used throughout this paper and spell out the details of our workload models. In addition, we discuss the simulation engine used for our studies and review some TCP-specific features that are necessary to present and discuss the findings of our simulation experiments.

2.1 A simple networking configuration

All of the simulation experiments reported in this paper involve the simple network topology depicted in Figure 1. Our toy network consists of a single server (node 1), a set of low-speed clients (nodes $5 - 405$), and a number of links. The clients are connected to the network via $40 - 100$ Kbps links, the server has a 100 Mbps connection to the network, and two additional links (A, B) comprise the rest of the network. Link A is used to limit the capacity of the network to a value between 1.5 Mbps and 3 Mbps and represents, in fact, the bottleneck link in our simulation setup. The link B bandwidth is set at 100 Mbps, and we use this link to observe the dynamics of the traffic before it traverses the bottleneck link A. To gauge the impact of the buffer size in the router, we vary the number of buffer spaces available for buffering packets[1] in node 3 for link A; depending on the particular experiment, this number can vary anywhere from 10 to 1000. Note that this network configuration allows for no variability as far as delays, round-trip times and cross traffic are concerned. While all of these three aspects of realistic network traffic are known to be crucial for matching the variability inherent in measured Internet traffic [4], our focus in this paper is on an extreme case of networking homogeneity, where there is a single bottleneck through which all the traffic has to go and where all packets experience one and the same delay in the network, namely 1.4 seconds (to compare, we also experiment with a link delay of 0.14 seconds). We chose this admittedly unrealistic network configuration because we wanted to illustrate our findings in a setting that has been used in a number of previous studies of the dynamics of TCP (e.g., [12, 16, 17]).

2.2 Two different workload models

In contrast to our homogeneous network configuration, our workload models for the clients are heterogeneous in nature and allow for a spectrum of variability at the user level. On the one side of the spectrum, we deal with the case of *no* user level variability by considering 50 *infinite sources* that always have data to transfer for the duration of the entire simulation (that is, 4200

---

[1] Ns-2 allocates buffer space in terms of number of packets and not number of bytes.

2.3 Some TCP background

The simulation engine used throughout this study is *ns-2* (Network Simulator version 2) [1]. This discrete event simulator provides a rich library of modules such as UDP, different flavors of TCP, scheduling algorithms, routing mechanism, and trace collection support. For the purposes of this paper, we rely on the fact that *ns-2* comes with a thoroughly-checked and well-studied implementation of TCP. To fully appreciate this advantage of using *ns-2* for our experiments, we first review those features of TCP which are of particular interest for the purpose of this study. For a more comprehensive treatment of TCP, see for example [13].

TCP is one of the predominant transport layer protocols in the Internet, providing a connection-oriented, reliable, byte stream between a source (or sender) and a destination (or receiver). TCP provides reliability by splitting the data into numbered segments. To insure that each segment is transmitted reliably, TCP – among other precautions – maintains a timer, RTO, for the different segments. If no acknowledgment for the segment is received by the sender from the receiver within the timer period RTO, TCP assumes that the segment has been lost and retransmits it. Another way for TCP to detect losses is based upon duplicated acknowledgments. Since acknowledgments are cumulative, and since every segment that is received out of order (i.e., non-consecutive segment numbers) triggers an acknowledgment, TCP assumes that four duplicated acknowledgments indicate that a segment was lost. As before, TCP in this case retransmits a segment if it detects a lost segment.

Since each end-system has limited buffer space and the network has limited bandwidth, TCP provides end-to-end flow control using a sliding window protocol. As far as the buffer-limited end-systems are concerned, the sender computes a usable window which indicates how much data the receiver buffer can handle. To deal with the bandwidth-limited network, the sender maintains a congestion window of size CWND. The sender can transmit up to the minimum of CWND and the usable window. At the start of a TCP connection, the usable window is initialized to the buffer size of the receiver, and the congestion window is limited to one or two segments. Each received acknowledgment, unless it is a duplicate acknowledgment, is used as an indication that data has been transmitted successfully and allows TCP to move the usable window and to increase the congestion window. However, increasing the congestion window depends on the state of the TCP connection. If the connection is in slow start, it is increased exponentially, or more precisely, by one segment for every acknowledgment. If the connection is in congestion avoidance, it is increased linearly by the maximum of one segment per round-trip time. TCP switches from slow start to congestion avoidance if the size of the congestion window is equal to the value of a variable called the slow start threshold, or SSTHRESH, for short. If TCP detects a packet loss (which in today's Internet is interpreted as an indication of congestion), either via timeout or via duplicated acknowledgments, SSTHRESH is set to half of the minimum of CWND and the usable window size. In addition, if the loss was detected via timeout, the congestion window is set to one segment.

3 Impact of variability at the application layer

In this section we demonstrate how the TCP flow control algorithm can lead to artifacts in the router buffer dynamics at node 3, depending on which of the two workload models are used to generate the network traffic. To this end, we compare the scenario where TCP only has to handle long-lived connections or "elephants" with that where it is faced with a mixture of "elephants" and short-lived "mice."

3.1 No variability and high synchronization

We first consider the case where 50 clients generate traffic according to the infinite source workload model and show in the left plot of Figure 2 some typical features associated with having 50 long-lived TCP connections or "elephants" responsible for all traffic seen on the network. More specifically, the left plot shows (i) the traffic rate process (i.e., number of packets per second) as it arrives at the queue at node 3, to be forwarded through the bottleneck link A to the different destinations, and (ii) the instantaneous buffer occupancy of the queue at node 3. We assume that the maximum buffer occupancy of the queue at node 3 is 50 packets and that the queueing discipline is "drop-tail", i.e., whenever an arriving packet sees a full buffer, it is dropped by the router. Each packet drop is viewed by TCP as an indication of network congestion and results in a signal back to the sender of the dropped packet to slow down its sending rate, which in turn reduces the overall packet rate process arriving at the node 3 queue. This feedback dynamic is an inherent feature of TCP's end-to-end flow control mechanism, and as far as the case of "elephants" is concerned, Figure 2 illustrates the effect of this dynamic on the rate process and the buffer occupancy. Shortly after the queue is filled and packets are dropped, the rate process stops increasing and drops after some time from a maximum of around 215 packets per second to about 150 packets per second. Such a drastic reduction in the rate process arriving at the queue allows the queue to completely drain its buffer. Upon experiencing no dropped packets, the individual TCP connections start to increase their sending rates again, which in turn results in an increase of the overall packet rate process (from about 150 to 215 or so packets per second) as seen by the queue. As a direct result of this higher arrival rate, the buffer at the node 3 queue starts to fill up again. Once the buffer is full, another round of packet drops will cause the affected connections to again reduce their sending rates, and the same process as before repeats itself. These arguments explain the predominant periodic fluctuations of the packet rate process and the buffer occupancy process depicted in the left plot of Figure 2, and the results are fully consistent with some of the original studies of the dynamics of TCP as described, for example, in [12] [17].

To demonstrate that these pronounced synchronization effects for the packet rate and buffer occupancy processes are not an artifact of the extremely large delay value of 640 milliseconds for link C, we show in the right plot of Figure 2 the results for of the same simulation experiment, except that the link C delay is now reduced to 40 milliseconds. Note that we still observe pronounced periodic behavior, but since the feedback to the clients is now much more immediate, the queue does not have time to drain completely, the cycle length is significantly shorter, the overall link utilization is larger, and the rate process is somewhat less variable. Nevertheless, the arguments for the observed synchronized behavior remain the same as before: many of the "elephants" experience multiple packet drops within a short period of time and are thus forced into into slow-start; from this point on, they proceed more or less in lock-step.

3.2 High variability and no synchronization

To demonstrate how allowing for realistic variability at the application level changes the qualitative features of Figure 2, we replace the infinite sources by clients that generate traffic according to our Web workload model. Intuitively, while retaining a few "elephants," this source model ensures that a significant portion of the TCP connections are short-lived (i.e., representing many "mice"). The results are shown in Figure 3, where we again depict the packet rate and buffer occupancy processes for the case of a 640 milliseconds (left plot) and 40 milliseconds (right plot) link C delay, respectively. In stark contrast to the visually obvious synchronization effects in the case of the infinite sources, the inherent variability of the TCP connections in the case of the Web sources results in a complete lack of any synchronization effects.

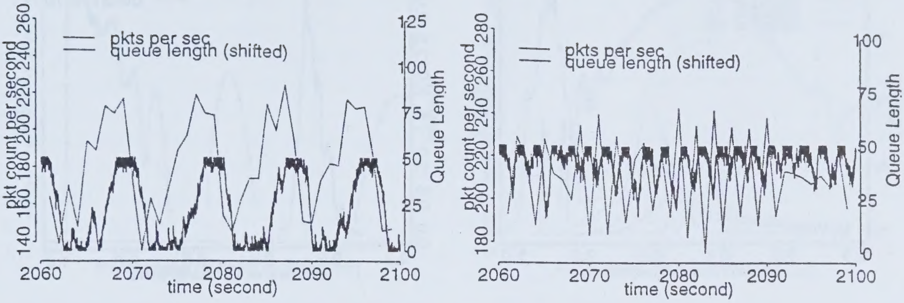


Figure 2: Packet rate process and buffer occupancy process (left: 50 infinite sources, link *C* delay 640 milliseconds; right: 50 infinite sources, link *C* delay 40 milliseconds).

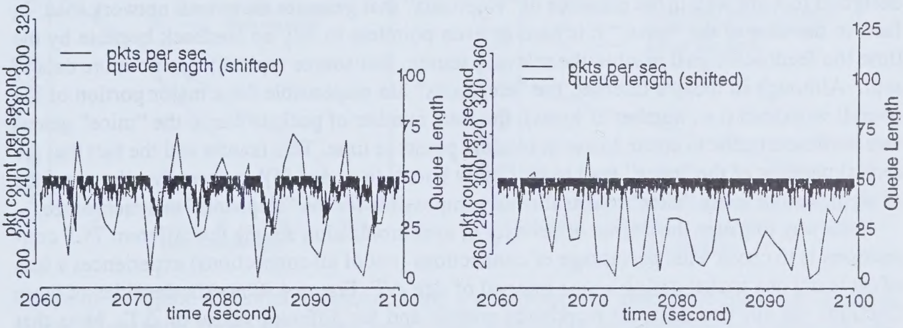


Figure 3: Packet rate process and buffer occupancy process (left: 350 PARETO 1-type Web clients, link *C* delay 640 milliseconds; right: 350 PARETO 1-type Web clients, link *C* delay 40 milliseconds).

The difference between the two workload scenarios is especially striking for the case of a link *C* delay of 640 milliseconds. While in the presence of many "mice", the buffer never has a chance to completely drain, it does so on a regular basis for the "elephants-only" case. Also note that while in the case of the Web sources, the packet rate process arriving at the node 3 queue never drops below 190 or so, it drops to about 140 in the case of infinite sources. Together, these effects result in a significantly higher overall utilization when there exists a proper mixture of "mice" and "elephants.". In the "elephants-only" case, even if we were to increase the number of long-lived connections, because of the presence of synchronization caused by the interaction between this type of workload and TCP feedback, the overall link utilization would not increase significantly.

3.3 On why "mice" can get rid of synchronization

In contrast to the infinite source model, our Web workload model – via its built-in Pareto-type distributions for the different Web session attributes – guarantees that a significant amount of TCP connections are very small and hence short-lived. However, TCP's feedback-based con-

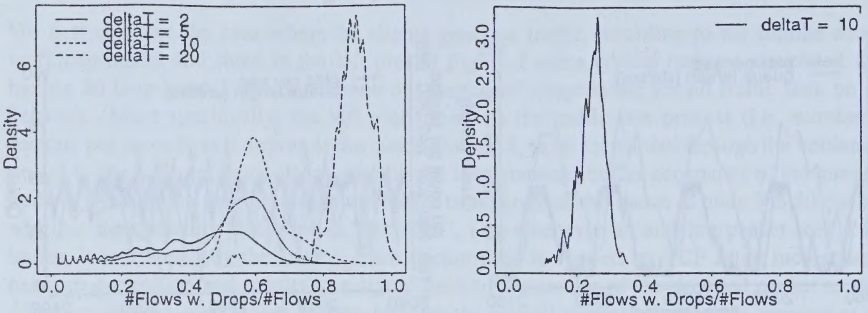


Figure 4: Effect of synchronization: Fraction of connections that experience packet losses.

gestion control mechanism was not intended for dealing with all these "mice" but was instead designed to work well in the presence of "elephants" that generate the overall network load. In fact, in the case of the "mice," it is hard or even pointless to rely on feedback because by the time the feedback signal reaches the relevant source, that source typically has no more data to sent. Although in today's Internet, the "elephants" are responsible for a major portion of the overall workload (i.e., number of bytes), the total number of packets due to the "mice" generates sufficient traffic to create losses at random points in time. This feature and the fact that the arrival patterns of the "mice" tend to be highly bursty (e.g., see [5]) suggest that the presence of significantly many "mice" makes it nearly impossible for the "elephants" to synchronize.[3]

One way to gauge the degree or severity of synchronization among the different TCP connections is to check what percentage of connections (out of all connections) experiences a loss of (at least) one packet during a time interval of size ΔT. Figure 4 shows the densities of these fractions for our two different workloads models and for different values of ΔT. Note that for the infinite source model (left plot), as ΔT increases from 1 second to 5 and 10 seconds, the densities become more centered around 60%, and when ΔT is increased even more to 20 seconds, the corresponding density function shifts toward 100%, with a mean larger than 90%. Thus, the TCP states of more than 50% of the connections can, in general, be assumed to be very similar. Moreover, given that the buffer occupancy process in Figure 2 (left plot) has a periodicity of about 10 seconds, we can conclude that about 60% of all connections lose at least one packet within this period and that almost every connection loses at least one of its packets within two such periods. Thus, even if a connection managed to avoid dropping a packet during one congestion epoch, it is almost certain to experience a packet drop during the subsequent cycle. In contrast, for the Web workload model (right plot), the density functions corresponding to the different ΔT values turn out to be essentially identical to one another (the plot only shows the density corresponding to $\Delta T = 10$), and they are all sharply concentrated around 25%. To explain this difference, note that "mice" can start when other connections are reducing their sending rates, and they often finish before they experience any dropped packet. This also explains why the fraction of connections experiencing packet drops is significantly smaller than in the absence of any "mice."

For another way to illustrate how the presence of many "mice" manifests itself in the observed TCP dynamics, we consider all connections that had (at least) one of their packets dropped and depict in Figure 5 the percentage p of those connections that were in slow-start

---

[3] Another known cause that works against the presence of synchronization effects in real network traffic is the observed variability in round-trip time [4], but this cause is not accounted for in our simulation setup.

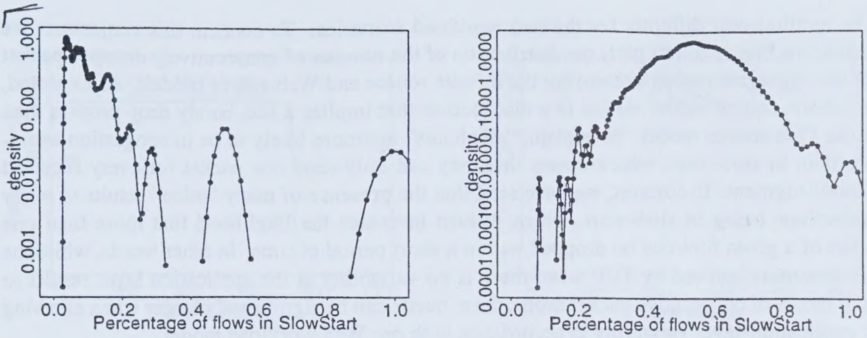


Figure 5: Effect of synchronization: Fraction of connections in slow-start.

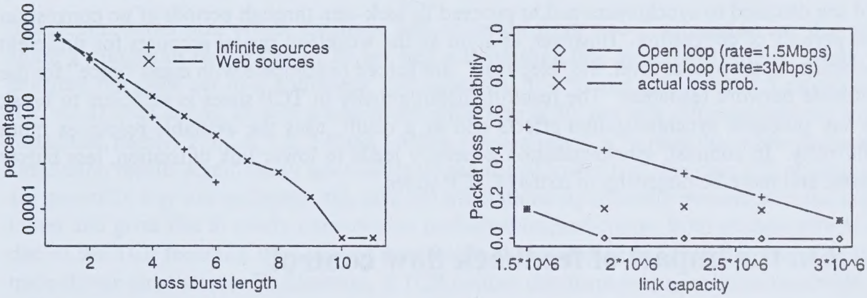


Figure 6: Left plot: Effect of synchronization – distribution of number of consecutive packet drops. Right plot: Open vs. closed loop – loss probability as a function of bottleneck link capacity.

at the time of packet drop (note that $1 - p$ gives the percentage of connections in congestion avoidance); the left plot is for the infinite source model, the right plot shows the same information for the Web source model.[4] The differences between the plots are again telling. In the presence of many "mice" (right plot), most of the connections that had any packet dropped are in slow-start, meaning that they are either very short or experience multiple dropped packets; a closer look (not shown here) reveals that both cases generally contribute to this dominant slow-start effect. In any case, with many of the connections going through slow-start, thereby increasing their bandwidth usage much more aggressively than those that are in congestion avoidance, the Web sources generally succeed via the presence of the many "mice" to claim any unused bandwidth and utilize the network resources efficiently. In contrast, in the absence of any "mice" (left plot), lots of the affected connections are in congestion avoidance and increase their bandwidth usage more gradually. Indeed, if we consider (not shown here) the size of the congestion window when losses occur for the infinite source case, it turns out to be significantly larger than when the sources generate traffic according to our Web workload model.

As a result of these observations, it is reasonable to expect the dynamics of packet drops to be qualitatively different for the two workload scenarios. To confirm this conjecture, we consider in Figure 6 (left plot) the distribution of the number of consecutively dropped packet (for the aggregate packet stream) for the infinite source and Web source models. As expected, the infinite source model results in a distribution that implies a less bursty drop process than for the Web source model. To explain, "elephants" are more likely to be in congestion avoidance than in slow-start, which means that they can only send one packet for every received acknowledgment. In contrast, we have seen that the presence of many "mice" results in many connections being in slow-start, which in turn increases the likelihood that more than one packet of a given flow can be dropped within a short period of time. In other words, while the loss dynamics induced by TCP when there is no variability at the application layer results in small bursts of consecutive packet drops, these bursts can be significantly larger when allowing for application-layer variability in accordance with our Web workload model.

---

[4] These percentages are calculated for successive time intervals of length two seconds, ignoring intervals without any dropped packets. Different choices for the length of the time interval yields essentially identical plots.

To summarize, our simulation experiments have demonstrated that the dynamics of TCP can interact in intricate ways with the dynamics of the underlying workload model for the sources. Static workloads such as infinite sources allow for no variability at the source level and are destined to synchronize and to proceed in lock-step through periods of no congestion and periods of congestion. However, as soon as the workload model accounts for sufficient variability at the source level, the "elephants" are forced to compete with many "mice" for the available network resources. The resulting heterogeneity in TCP states is sufficient to break up any potential synchronization effects and as a result, uses the available resources more efficiently. In contrast, synchronization generally leads to lower link utilization, less bursty losses, and more homogeneity in terms of TCP states.

4 On the impact of feedback flow control

In the following, we briefly describe some of the results of a second set of *ns-2* simulation experiments that was intended to demonstrate how the network through the TCP end-to-end congestion control algorithm shapes (with some delay) the packet flow emitted from the different sources, which in turn alters the rate process that arrives at the IP router for buffering (which in turn impacts the level of congestion, etc.). To this end, we performed a number related *closed loop* and *open loop* simulations and compared them on the basis of some commonly-used performance criteria. Here, by "closed loop" we mean a *ns-2* simulation with a fixed simple topology, including buffer size in the router(s), link bandwidths, delays, etc. and where all hosts use TCP. In contrast, "open loop" means we collect a packet trace from a particular *ns-2* simulation run and use it to perform a trace-driven simulation of a queueing system that represents our IP router. Note that the open loop nature of trace-driven simulations cannot account for the capabilities of the network to shape and thus alter the offered traffic to the queue (e.g., as a result of changing congestion levels in the network through, say, increasing the buffer in the router or by means of changing the capacity of the bottleneck link).

To illustrate, a commonly used method for investigating the buffer dynamics in a queue with constant service time is to consider an actual packet-level traffic trace, collected from some link in, say, the Internet, and use it as input for a number of open loop trace-driven simulations. For example, by changing the bandwidth of the output link of the queue, we can change the service or drain rate of the queue and, as result, study the packet loss probability as a function of the (bottleneck) link bandwidth. The right plot in Figure 6 shows the results of (i) a set of 5 open loop trace-driven simulations (maximum buffer size of 50; the trace was collected from a *ns-2* simulation run that used a maximum buffer size of 50 and a bottleneck link speed of 1.5 Mbps), where we fixed the bottleneck link capacity (i.e., link A) at $1.5, 1.8, 2.2, 2.6$ and 3 Mbps, respectively ("$\diamond$"); (ii) another set set of 5 open loop trace-driven simulations

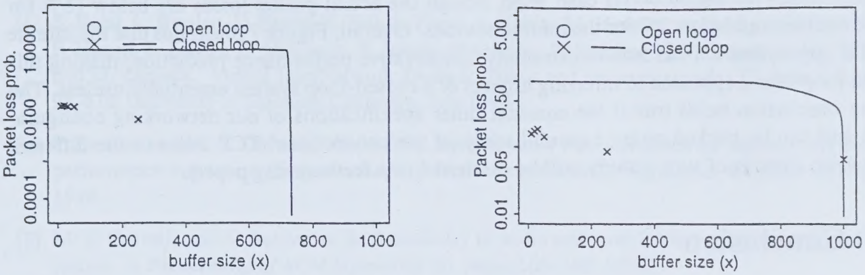


Figure 7: Open loop vs. closed loop, infinite sources (left) vs. Web sources (right): Loss probability vs. buffer size.

(maximum buffer size of 50; the trace was collected from a *ns-2* simulation run that used again a maximum buffer size of 50 but had a bottleneck link speed of 3 Mbps), where we consider the same set of a bottleneck link capacities ("+"); and (iii) a set of 5 closed loop (i.e., *ns-2*) simulations, all with a maximum buffer size of 50, but with different bottleneck link capacities, namely 1.5, 1.8, 2.2, 2.6 and 3 Mbps, respectively ("×"). Note that while the case (i) simulation results significantly underestimate the actual losses (i.e., the case (iii) results) and are generally way too optimistic, the case (ii) simulations significantly overestimate the actual losses and gives rise to overly conservative performance predictions. Both of these effects are due to the TCP feedback flow control mechanism that is explicitly ignored in the open loop trace-driven simulations. To illustrate, if TCP notices that there exists available bandwidth, it will in general allow the individual connections to increase their sending rates. On the other hand, if TCP receives too many indications of network congestion (via dropped packets), it has mechanisms in place that ensure that the offered load will be reduced. Similar results (not shown here) hold for performance measures other than loss probability, or when varying, for example, the maximum buffer size instead of the bottleneck link capacity.

Finally, we comment on another commonly-used approach for inferring buffer dynamics in an Internet-like setting from open loop trace-driven simulations, where in addition, the workload model comes into play. It is common engineering practice to use the the complementary probability distribution of the buffer occupancy in an infinite buffer queue as an accurate substitute for the loss probability in a finite buffer system. To check the validity of this practice in our simple networking setting, we run a number of identical *ns-2* simulations, except that we considered different maximum buffer occupancies at node 3, namely 10, 20, 30, 50, 250 and 1000, and obtained the actual loss probabilities as a function of the buffer size at the node 3 queue. As our infinite buffer system, we take the simulation with maximum buffer size of 1000 and infer from it the complementary probability distribution function that the buffer exceeds a certain value x. The results are depicted in Figure 7 (the solid lines correspond to the infinite buffer approximation, while the crosses indicate the actual loss probabilities), where the left plot is for the infinite source case and the plot to the right is based on the Web sources. For the infinite sources and a maximum buffer size of 1000, the queue length process resulting from the *ns-2* simulation run (i.e., closed loop) turns out to tightly fluctuate around 725. In fact, the maximum buffer capacity is never exceeded, resulting in no losses and implying that in this scenario, the sources are bandwidth limited on their access links. In contrast, the *ns-2* simulation for the 1000 buffer size case with Web sources does occasionally fill the whole buffer resulting in significant losses of more than 5%. On the other hand, note that the infinite buffer approximations are by definition open loop based and predict, for example, almost 100% packet losses for the 50 buffer case, even though the actual packet losses are below 15% for Web sources and below 3% for the infinite sources. Overall, Figure 7 illustrates that the infinite buffer approximation can lead to extremely conservative performance prediction, making this open loop-based approach to inferring aspects of a closed-loop system essentially useless. The same conclusion holds true if we consider other specifications of our networking configuration and can be backed up by a careful study of the connections' TCP states in the different scenarios (details of such a study will be discussed in a forthcoming paper).

5 Conclusion

Even though the networking configuration considered in our simulation experiments is admittedly unrealistic and oversimplified, based on our findings, a word of caution is in place when generalizing results obtained in environments with infinite sources and open loop systems to real networks such as the Internet, where a major part of the traffic is generated by Web users and uses TCP. In fact, we have illustrated with a few examples that such generalizations can lead to highly conservative if not meaningless performance predictions. While infinite source models and open loop toy examples can provide deep insight into and physical understanding of the dynamics of real networks, we believe that their credibility could be substantially enhanced by demonstrating that the insight and understanding they provide (i) remain essentially unchanged, or (ii) may require appropriate modifications, or (iii) are no longer applicable when accounting for realistic variability at the source level and when taking the impact of closed loop feedback controls into serious consideration.

Looking ahead, it will be interesting to see whether or not some of the generic differences observed in our abstract setting will remain valid for more realistic network configurations. Another less obvious shortcoming of our experiments presented in this paper is that we completely ignore the potential of feedback from the network all the way back to the application layer; that is, the congestion state of the network may have a direct impact on our Web-user source model because it may directly influence the Web-browsing behavior of individual users. While there exists mainly anecdotal evidence for the presence of such types of feedback behavior (e.g., Internet "storms" [8]), we have seen little empirical evidence for the widespread existence of such feedback in our analysis of a wide variety of Internet traffic measurements. Nevertheless, the potential pitfalls associated with assuming an open loop characterization at the source level should be kept in mind and may require revamping the current approach to source modeling, depending on how the Internet develops in the near future. Other aspects not considered in our experimental studies concern replacing the drop-tail queueing discipline in the router by, say, RED, for "random early drop" [7]; incorporating TCP features such as SACK (selected ack) or delayed acks [10]; and dealing with the problem of two-way or cross traffic (e.g., see [17, 4]). Part of our ongoing efforts to understand the dynamics of TCP traffic in a realistic networking setting deals with some of these aspects and how they impact our current understanding, and will be published elsewhere.

Acknowledgments

We would like to acknowledge Polly Huang for her help with the *ns*-2-simulations.

References

[1] S. Bajaj, L. Breslau, D. Estrin, K. Fall, S. Floyd, P. Haldar, M. Handley, A. Helmy, J. Heidemann, P. Huang, S. Kumar, S. McCanne, R. Rejaie, P. Sharma, S. Shenker, K. Varadhan, H. Yu, Y. Xu, and D. Zappala. Virtual InterNetwork Testbed: Status and research agenda. Technical Report 98-678, University of Southern California, July 1998.

[2] P. Barford and M. E. Crovella. Generating representative web workloads for network and server performance evaluation. In *Proceedings of Performance'98/ACM Sigmetrics'98*, pages 151–160, 1998.

[3] M. E. Crovella and A. Bestavros. Self-similarity in world wide web traffic - evidence and possible causes. In *Proceedings of ACM Sigmetrics'96*, pages 160–169, 1996.

[4] A. Feldmann, A. C. Gilbert, P. Huang, and W. Willinger. Dynamics of IP traffic: A study of the role of variability and the impact of control. In *Proc. of the ACM/SIGCOMM'99*, pages 301–313, Boston, MA, 1999.

[5] A. Feldmann, A. C. Gilbert, and W. Willinger. Data networks as cascades: Investigating the multifractal nature of Internet WAN traffic. In *Proc. of the ACM/SIGCOMM'98*, pages 25–38, Vancouver, B.C., 1998.

[6] A. Feldmann, A. C. Gilbert, W. Willinger, and T. G. Kurtz. The changing nature of network traffic: Scaling phenomena. *Computer Communication Review*, 28(2):5–29, 1998.

[7] S. Floyd and V. Jacobson. Random early detection gateways for congstion avoidance. *IEEE/ACM Transactions on Networking*, 1(4):397–413, 1993.

[8] B. A. Huberman and R. M. Lukose. Social dilemmas and Internet congestion. *Science*, 277:535–537, 1997.

[9] V. Jacobson. Congestion Avoidance and Control. In *Proc. of the ACM/SIGCOMM'88*, pages 314–329, 1988.

[10] V. Jacobson, R. Braden, and D. Berman. Tcp extensions for high performance, May 1992. Request for Comments 1323.

[11] V. Paxson and S. Floyd. Wide area traffic: The failure of poisson modeling. *IEEE/ACM Transactions on Networking*, 3:226–244, 1995.

[12] S. Shenker, L. Zhang, and D. Clark. Some Observations on the Dynamics of a Congestion Control Algorithm. *ACM Computer Communication Review*, 20(4):30–39, October, 1990.

[13] W.R. Stevens. *TCP/IP Illustrated Volume 1*. Addison-Wesley, 1994.

[14] W. Willinger, V. Paxson, and M. S. Taqqu. *A Practical Guide to Heavy Tails: Statistical Techniques for Analyzing Heavy Tailed Distributions*, chapter Self-similarity and heavy tails: Structural modeling of network Traffic. Birkhauser Verlag, Boston, 1998.

[15] W. Willinger, M. S. Taqqu, R. Sherman, and D. V. Wilson. Self-similarity through high-variability: Statistical analysis of Ethernet LAN traffic at the source level. *IEEE/ACM Transactions on Networking*, 5:71–86, 1997.

[16] L. Zhang and D. Clark. Oscillating Behavior of Network Traffic: A Case Study Simulation. *Internetworking: Research and Experience*, 1(2):101–112, 1990.

[17] L. Zhang, S. Shenker, and D. Clark. Observations on the Dynamics of a Congestion Control Algorithm: The Effects of Two-Way Traffic. In *Proc. of the ACM/SIGCOMM'91*, pages 133–147, 1991.

Queueing Networks with Interacting Service Resources

Mor Armony
Department of Operations Management
Stern School of Business
New York University
New York, NY 10012-1118
marmony@stern.nyu.edu

Nicholas Bambos
Department of Engineering-Economic Systems and Operations Research
Stanford University
Stanford, CA 94305-4023
bambos@stanford.edu

Abstract

We study multiclass queueing systems, whose service facility can be in one of M configurations: $\vec{R}_1, .., \vec{R}_m, .., \vec{R}_M$. When it is in configuration m, the service rate dedicated to class q is R_m^q. The main issue is how to dynamically control the service configuration to optimize various performance measures. In this paper, we study two general families of processing policies, that maximize system throughput and address important practical considerations seen in a number of motivating applications.

1 Introduction

The service mechanism in multiclass queueing networks has traditionally been assumed to be such that the capacity of the servers is shared among several customer classes in a "linear" fashion. In other words, an increase in the service effort devoted to one class, results in an equivalent decrease in the total service effort devoted to the other classes. Recent developments in Information Technologies and Service Operations have given rise to more complex systems, that have service resources, which interact in a highly non-linear manner. In this paper we introduce a model which allows one to capture very general interdependencies among the service resources within a unifying, simple framework.

Throughout this paper we consider feed-forward (acyclic) queueing networks, with nodes that include three basic components; the arriving jobs, the buffer, and the service mechanism. Jobs arriving to a node are assumed to be assigned to one of Q parallel, infinite buffer queues. The Q job classes are distinguished by different statistical properties such as inter-arrival time and service requirement distributions. The service mechanism can operate according to one of M configurations at any point in time. When configuration m is used, the $q-$th queue receives service rate R_m^q. The service configurations are allowed to be as general as desired and, therefore, almost any level of resource interaction can be captured by this model. In this paper we focus on the analysis of such *basic nodes*, but note that most results can be extended to feed-forward networks of interconnected nodes of this sort.

There are numerous applications that can be studied through the analysis of the basic node. The following are a few examples from the areas of manufacturing, information services, and telecommunication systems:

Scheduling jobs and tools in flexible manufacturing: Processing different jobs requires the use of specific tools, all taken from a common pool.

Skilled based routing in call centers: Calls can be handled by various servers that are differentiated by skills (service rates).

Power control in cellular systems: Transmitters interfere with each other when transmitting in the same channel. Changing the transmission power changes the level of interference, and thereby alters the rates of successful transmissions.

Dynamic scheduling in switching systems with shared input and output ports: The control here is on the set of possible matchings between the input and the output ports. Each such matching determines the classes of packets to be processed.

Once a system has been formulated as a basic node (or as a feed-forward network of such nodes), we focus our attention on how to dynamically control the service configuration to optimize the system's performance; namely, to maximize throughput and minimize operational costs. The structural complexity of such systems requires the control policies to be simple and easy to implement. There are other considerations that need to be taken into account, such as: real time response to changes in the load of the system, low complexity (overhead of computation), scalability, availability of information at decision epochs, non-preemption, etc. In some cases, there is also a switching (setup) cost involved in changing from one configuration to the other. In such systems the frequency of configuration switchings should be relatively low. Various families of control policies are studied, that are designed to address these issues.

The main performance criterion that is studied in this paper is throughput maximization; the *stability region* is expressed as a simple function of the parameters of the system, and throughput maximizing policies are identified, that stabilize the system, as long as the parameters are in this region. The probabilistic framework under which this problem is studied, is *rate ergodicity* of the input process. That is, we assume a strong law of large numbers for the rate of workload that arrives into the system. This is a very mild assumption in practice, but at the same time is sufficiently restrictive to permit the verification of statements regarding the system's stability. The form of stability that we consider is *rate stability* (arrival rate = departure rate). Although rate stability is, in general, a weaker condition than other forms of stability, it is the natural stability concept within the rate ergodicity framework.

The model formulation in this paper is motivated by the one studied by Bambos and Walrand in [1], in the context of parallel processing systems. This model is further analyzed by Gans and van Ryzin in [3]. There, they give performance bounds for a selection of policies, under renewal assumptions. Special cases of the cone policies discussed in this paper appear in various publications. In particular, Wasserman and Lennon-Olsen in [8] study such policies in the power control context, McKeown, Anantharam and Walrand, in [6], use them for High Speed Switches, and Tassiulas and Ephermides apply them to Multihop Radio Networks (see [7]). The discussion of adaptive batching in this paper is also motivated by Bambos and Walrand [1], but admits a broader scope here, both in terms of applications, as well as the variety of policies that it refers to. The idea of adaptive batching, or discrete review policies, has gained a lot of attention in the literature. In particular, examples can be found in Gans and van Ryzin [3], Harrison [4], and Maglaras [5].

1.1 Notation

Suppose that jobs arrive to any of the Q parallel queues randomly according to the input flows $\mathbb{I}^q = \{(t^q_j, \sigma^q_j),\ j \in \mathbb{Z}\}$, $q = 1,..,Q$, where t^q_j is the arrival time of job j into queue q, and σ^q_j is its service requirement. The implied service time of job j is $\frac{\sigma^q_j}{r}$, provided that r is the service rate used when processing this job. The input flows $\{\mathbb{I}^q : q = 1,..,Q\}$ are defined on some common probability space, $\{\Omega, \mathcal{F}, P\}$. All the input flows are assumed to be *rate ergodic* with respect to the transformation: $\theta_z \mathbb{I}^q = \{(t^q_j - z, \sigma^q_j),\ j \in \mathbb{Z}\}$, which essentially means that input rates exist almost surely, and hence, are well defined. In particular, rate ergodicity implies the existence of the *traffic intensity* into queue q, which is defined as the average workload that arrives into this queue per unit of time, and is given by:

$$\rho^q = \lim_{t\to\infty} \frac{\Sigma^q(s,t)}{t-s} = \lim_{s\to-\infty} \frac{\Sigma^q(s,t)}{t-s},$$

almost surely, for every fixed s and t, correspondingly. Here, $\Sigma^q(s,t)$ represents the total workload that arrives into queue q between the time points s and t, that is, $\Sigma^q(s,t) = \sum_{j\in\mathbb{Z}} \sigma^q_j \mathbf{1}_{\{t^q_j \in (s,t]\}}$.

Let $\left\{\vec{R}_1, \vec{R}_2, .., \vec{R}_M\right\} \subseteq \mathbb{R}^Q_+$ be the set of service configurations (vectors), where R^q_m is the service rate of server q, if the service vector $\vec{R}_m$ is used ($q = 1,..,Q,\ \ m = 1,..,M$). A processing policy is a set of rules that determine the service vector to be used at each time t, depending on the current state and the history of the system. Let Π be the set of all processing policies, and let $\pi \in \Pi$ be a given policy. For all $m = 1,..,M$, we define $T_m(s,t,\pi)$ to be the random cumulative time that $\vec{R}_m$ is used in the interval (s,t], according to the policy π. Clearly, for all $s < t$, $\sum_{m=1}^M T_m(s,t,\pi) \le t - s$. Suppose that at time s, the initial workload is some $\vec{w}_{init} \in \mathbb{R}^Q_+$, and that a policy $\pi \in \Pi$ is used. Then the workload process is some $\vec{W}(s,t,\pi)$ and its $q-th$ component is given by:

$$W^q(s,t,\pi) = w^q_{init} + \sum_{j\in\mathbb{Z}} \sigma^q_j \mathbf{1}_{\{t^q_j\in(s,t]\}} - \sum_{m=1}^{M} R^q_m \int_s^t \mathbf{1}_{\{\pi \text{ uses } \vec{R}_m \text{ at } \tau\}} \mathbf{1}_{\{W^q(s,\tau,\pi)>0\}} d\tau \qquad (1)$$

and hence, the workload process satisfies,

$$\vec{W}(s,t,\pi) = \vec{w}_{init} + \vec{\Sigma}(s,t) - \sum_{m=1}^{M} \vec{R}_m T_m(s,t,\pi) + \vec{Y}(s,t,\pi) \qquad (2)$$

where $\vec{Y}(s,t,\pi)$ is the potential outflow that is lost due to emptiness of the buffers. In other words, $\vec{Y}(s,t,\pi)$ is the minimal non-negative process that can be added to $\vec{w}_{init} + \vec{\Sigma}(s,t) - \sum_{m=1}^M \vec{R}_m T_m(s,t,\pi)$, in order to guarantee that $\vec{W}(s,t,\pi)$ stays non-negative (see [2]). In what follows, if $s = 0$, then s is omitted, and the policy π is omitted, if it is clear from the context which policy is used.

The rest of the paper is organized as follows: In section 2, we describe the stability region. Dynamic cone policies are described in section 3, where they are also shown to maximize system's throughput. In section 4, the adaptive batching approach is briefly described. We finally make some concluding remarks in Section 5.

2 The Stability Region

In this section we describe the *stability region*, namely, the set of all traffic intensity vectors $\vec{\rho}$, for which there exists a policy under which the system is stable. This region is expressed as a function of the service vectors: $\vec{R}_1$, $\vec{R}_2, .., \vec{R}_M$. As it turns out, this function is quite simple, and has a straightforward geometrical interpretation.

The form of stability that is considered in this paper is *rate stability*. This essentially means that the departure rate of jobs from the system is equal, in the long run, to the arrival rate. This equality is easily seen to be captured by the definition below:

Definition. The system is *rate stable* under a policy $\pi \in \Pi$, if the workload process satisfies: $\lim_{t\to\infty} \frac{\vec{W}(s,t,\pi)}{t} = 0$ *a.s.* $\forall s$, and any initial workload vector at time s.

Given this definition, the stability region corresponds to all the traffic intensity vectors $\vec{\rho}$, such that there *exists* a policy under which the system is rate stable.

Before characterizing the stability region, we consider the following different, but related, problem. Suppose that $\vec{W} \in \mathbb{R}_+^Q$ is the workload present in the system, and no additional arrivals are admitted. Consider the problem of emptying the system in the minimal possible time, using the service vectors: $\vec{R}_1$, $\vec{R}_2, .., \vec{R}_M$. Service preemption is allowed. This problem can be formulated as the following linear program:

$$\begin{aligned} &\textit{minimize} \quad \sum_{m=1}^{M} x_m \\ &\textit{subject to} \quad R\vec{x} \geq \vec{W} \\ &\qquad\qquad\quad \vec{x} \geq 0 \end{aligned} \qquad (LP(\vec{W}))$$

Here, R is the $Q \times M$ matrix whose columns are the service vectors $\vec{R}_1, .., \vec{R}_m, .., \vec{R}_M$. x_m, $m = 1, .., M$, corresponds to the time spent using service vector $\vec{R}_m$, while processing the workload $\vec{W}$. The optimal objective value is nothing but the minimal emptying time of the system. Let $\vec{x}^*(\vec{W})$ be an optimal solution of $LP(\vec{W})$, and let $\delta(\vec{W}) = \sum_{m=1}^{M} x_m^*(\vec{W})$ be the optimal objective value. In particular, let $\delta(\vec{\rho})$ be the minimal time to empty the system, if a workload of $\vec{W} = \vec{\rho}$ is initially present in the system. Note that $LP(\vec{W})$ has an optimal solution for all $\vec{W} \in \mathbb{R}_+^Q$, as the problem is feasible, and the value of the objective function is bounded below by 0.

Let $\left\{\vec{R}_1, \vec{R}_2, .., \vec{R}_M\right\} \subseteq \mathbb{R}_+^Q$ be the set of service vectors, and suppose that $\delta(\vec{\rho})$ is the optimal solution of $LP(\vec{\rho})$, for the traffic intensity vector $\vec{\rho}$. Let $\mathbb{S} = \left\{\vec{\rho} \in \mathbb{R}_+^Q \mid \delta(\vec{\rho}) \leq 1\right\}$, then $\mathbb{S}$ is the stability region. In other words, if the traffic intensity vector $\vec{\rho} \in \mathbb{S}$, then there exists a policy, under which the system is stable. On the other hand, if $\vec{\rho} \notin \mathbb{S}$, then the system is unstable, regardless of which policy is used. We now state the stability and the instability results formally in propositions 1 and 2. The proof of proposition 1 will be given in Sections 3 and 4, by describing throughput maximizing policies (i.e. policies under which the system is stable, whenever $\vec{\rho} \in \mathbb{S}$). Due to space limitations, we only give a sketch of the proof of Proposition 2.

Proposition 1. *Stability; Let* $\mathbb{I}^q = \left\{(t_j^q, \sigma_j^q),\ j \in \mathbb{Z}\right\}$, $q = 1, .., Q$, *be the input flows into the system, with traffic intensity vector* $\vec{\rho}$. *Suppose that* $\vec{\rho} \in \mathbb{S}$. *Then there exists a policy* $\pi \in \Pi$, *under which the system is rate stable, i.e.,* $\lim_{t\to\infty} \frac{\vec{W}(s,t,\pi)}{t} = 0$ *a.s.* $\forall s$.

Proposition 2. *Instability; Let* $\mathbb{I}^q = \{(t_j^q, \sigma_j^q),\ j \in \mathbb{Z}\}$, $q = 1, .., Q$, *be the input flows into the system, with traffic intensity vector* $\vec{\rho}$. *Suppose that* $\vec{\rho} \notin \mathbb{S}$. *Then the system is unstable, regardless of which policy is used. Moreover, there exists* $\epsilon > 0$, *such that for all* $\pi \in \Pi$, $\liminf_{t\to\infty} \frac{\sum_{q=1}^{Q} W^q(s,t,\pi)}{t} \geq \epsilon$ *a.s.*

Sketch of the proof: The proof follows the sample path approach, and the key idea is that, when $\rho \notin \mathbb{S}$, the incoming workload is more than what can be handled by the service mechanism. ∎

We conclude this section by giving a geometric interpretation of the stability region as a function of the service vectors. As stated in proposition 1, the stability region corresponds to the set $\mathbb{S}$ of all traffic intensity vectors $\vec{\rho}$, such that $\delta(\vec{\rho}) \leq 1$. It is easy to see that the set $\mathbb{S}$ is equivalent to the set of all vectors that are less than or equal to the convex hull of the service vectors. Lemma 1 states this formally.

Lemma 1. *Let* $\vec{\rho} \in \mathbb{R}_+^Q$. *Then* $\vec{\rho} \in \mathbb{S}$ *if and only if* $\vec{\rho}$ *is less than or equal to a convex combination of the service vectors:* $\vec{R}_1, \vec{R}_2, .., \vec{R}_M$.

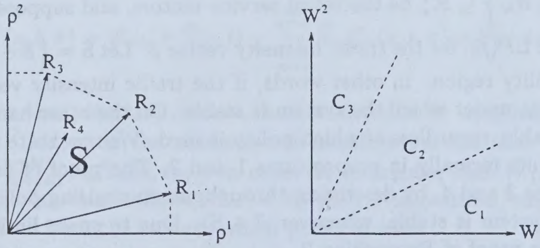
Proof: The proof follows directly from the definitions of $\delta(\cdot)$ and $\mathbb{S}$. ∎

3 Dynamic Cone Policies

In this section we describe dynamic cone policies that update the service vectors dynamically and continuously, according to the value of the workload realization at any time point. Such policies are particularly useful when the system under consideration needs real time response to changes in the loads. This issue is especially crucial in telecommunication systems, where traffic can be very bursty, and long delays can result in a significant loss of information.

A typical cone policy is characterized by a cone partitioning, $\{C_m \mid m = 1, .., M\}$, of the workload space (*i.e.* $\cup_{m=1}^{M} C_m = \mathbb{R}_+^Q$). Given such a partitioning, the service vector, $\vec{R}_m$, is used whenever $\vec{W}(t) \in C_m$. Note that, in general, some of the cones might be empty. Also note, that although we use the term: "cone partitioning", it might be the case that the intersection of two different cones is non-empty. If this is the case, one of the relevant service vectors may be chosen arbitrarily.

Figure 1 shows an example of a cone partitioning, in the case of $Q = 2$. Note that the cone corresponding to R_4 is empty.



The Stability Region

The Cone Partitioning

Figure 1. *A graphic illustration of a cone partitioning.*

In this paper, we restrict our attention to cone policies that can be characterized in the following way: Let $\vec{\alpha} \in \mathbb{R}_+^Q$. We define the $\vec{\alpha}$-dependent operator on $\mathbb{R}^Q \times \mathbb{R}^Q$ as follows: For two vectors, $\vec{u}, \vec{v} \in \mathbb{R}^Q$, let

$$\langle \vec{u}, \vec{v} \rangle_{\vec{\alpha}} := \sum_{q=1}^{Q} \alpha^q u^q v^q.$$

Clearly, if $\vec{\alpha}$ is strictly positive, then $\langle \cdot, \cdot \rangle_{\vec{\alpha}}$ is an inner product on $\mathbb{R}^Q \times \mathbb{R}^Q$. Next, we define the cone partitioning for a fixed positive vector, $\vec{\alpha} \in \mathbb{R}_+^Q$. For $m = 1, .., M$, let

$$C_m = \left\{ \vec{W} \in \mathbb{R}_+^Q \mid \left\langle \vec{W}, \vec{R}_m \right\rangle_{\vec{\alpha}} = \max_{m'=1,..,M} \left\langle \vec{W}, \vec{R}_{m'} \right\rangle_{\vec{\alpha}} \right\}$$

It can be easily verified that for $m = 1, .., M$, C_m is indeed a cone. It is also clear that $C_m \neq \phi$ if and only if $\vec{R}_m$ is on the "north-east" boundary of the stability region. Finally, from the definition of the cones, it is evident that $\left\langle \vec{W}, \vec{R}_m \right\rangle_{\vec{\alpha}}$ can potentially have more than one maximizer. In such cases, the service vector can be chosen arbitrarily, from the set of the maximizers. For a given $\vec{\alpha}$, and a workload vector, $\vec{W}$, we define $A(\vec{W})$ as the set of maximizers of the inner product. That is,

$$A(\vec{W}) = \left\{ m \mid m = argmax_{m'=1,..,M} \left\langle \vec{W}, \vec{R}_{m'} \right\rangle_{\vec{\alpha}} \right\}.$$

Note, that although the dynamic cone policies are simple in structure, they provide a rich selection of continuously updated policies, depending on the choice of the vector $\vec{\alpha}$. This variety allows one to choose the appropriate weights, α^q, $q = 1, .., Q$, for the different queues, so as to create weighted class priorities, depending on various design considerations and performance specifications. Also note, that these policies do not require the knowledge of the traffic intensity vector, and that it is sufficient to know the service vectors and the specific choice of $\vec{\alpha}$, as well as the value of the workload realization at each time point, when implementing them.

The following lemma describes a basic inequality, and proves to be useful later on.

Lemma 2. *Suppose that $\vec{\rho} \in \mathbb{S}$, and let $\vec{W} \in \mathbb{R}_+^Q$ be an arbitrary non negative vector. Also, suppose that $\{f_m\}_{m \in A(\vec{W})}$, are non negative numbers such that $\sum_{m \in A(\vec{W})} f_m = 1$, then*

$$\left\langle \vec{W}, \vec{\rho} - \sum_{m \in A(\vec{W})} f_m \vec{R}_m \right\rangle_{\vec{\alpha}} \leq 0 \tag{3}$$

Proof: The result follows from Lemma 1 ∎

The rate stability of the dynamic cone policies is stated next.

Proposition 3. *For a fixed positive vector, $\vec{\alpha} \in \mathbb{R}_+^Q$, let π be the dynamic cone policy based on $\langle \cdot, \cdot \rangle_{\vec{\alpha}}$. If $\vec{\rho} \in \mathbb{S}$, then the system is rate stable under π.*

Before proving the proposition, we formulate and prove three additional results. In what follows, $\vec{\alpha} \in \mathbb{R}_+^Q$ is a fixed positive vector.

Claim 1. *Suppose that for some increasing, unbounded, sequence of time points $\{t_n\}_{n=1}^{\infty}$, $\lim_{n\to\infty} \frac{\vec{W}(t_n)}{t_n} = \vec{\eta}$, for some $\vec{\eta} \in \mathbb{R}_+^Q$. Let m_0 be such that $m_0 \notin A(\vec{\eta})$ (assuming that such m_0 exists). Let $\tau_n < t_n$, such that $m_0 \in A(\vec{W}(\tau_n))$ (if there is no such time point, let $\tau_n = 0$). Then, $\liminf_{n\to\infty} \frac{t_n - \tau_n}{t_n} > 0$.*

Proof: Suppose, by contradiction, that, with positive probability, $\liminf_{n\to\infty} \frac{t_n-\tau_n}{t_n} = 0$. Without loss of generality, assume that $\lim_{n\to\infty} \frac{t_n-\tau_n}{t_n} = 0$. This in particular means that,

$$\lim_{n\to\infty} \frac{\vec{\Sigma}(\tau_n, t_n)}{t_n} = \lim_{n\to\infty} \frac{\vec{\Sigma}(0, t_n)}{t_n} - \lim_{n\to\infty} \frac{\vec{\Sigma}(0, \tau_n)}{\tau_n}\frac{\tau_n}{t_n} = \vec{\rho} - \vec{\rho} = 0.$$

Recall that $T_m(t)$ corresponds to the cumulative time spent using service vector $\vec{R}_m$ up to time t, and that $\vec{Y}(t)$ is the cumulative potential outflow lost due to emptiness of the system, up to time t. Then the assumption that $lim_{n\to\infty} \frac{t_n-\tau_n}{t_n} = 0$, also implies that

$$0 \leq \lim_{n\to\infty} \frac{T_m(t_n) - T_m(\tau_n)}{t_n} \leq \lim_{n\to\infty} \frac{t_n - \tau_n}{t_n} = 0, \quad m = 1, .., M,$$

and

$$0 \leq \lim_{n\to\infty} \frac{\vec{Y}(t_n) - \vec{Y}(\tau_n)}{t_n} \leq \lim_{n\to\infty} \frac{\sum_{m=1}^{M} \vec{R}_m(T_m(t_n) - T_m(\tau_n))}{t_n} = 0,$$

which together with (2) imply that,

$$\lim_{n\to\infty} \frac{\vec{W}(\tau_n)}{t_n} = \vec{\eta}$$

This, in particular, implies that there exists an N, such that for all $n > N$, $m_0 \notin A(\frac{\vec{W}(\tau_n)}{t_n})$, and hence, $m_0 \notin A(\vec{W}(\tau_n))$, which contradicts the definition of τ_n. $\blacksquare$

Claim 2. *Suppose that for some increasing, unbounded, sequence of time points* $\{t_n\}_{n=1}^{\infty}$, $\lim_{n\to\infty} \frac{W(t_n)}{t_n} = \vec{\eta}$, *for some* $\vec{\eta} \in \mathbb{R}_+^Q$, *with* $\eta^q > 0$ *for some* $q \in \{1, .., Q\}$. *Let* $\tau_n^q = \sup\{t < t_n \mid W_t^q = 0\}$ ($\tau_n^q = 0$, *if* $W_t^q > 0$, $\forall t < t_n$). *Then,* $\liminf_{n\to\infty} \frac{t_n-\tau_n^q}{t_n} > 0$.

Proof: Suppose, by contradiction, that, with positive probability, $\liminf_{n\to\infty} \frac{t_n-\tau_n^q}{t_n} = 0$. Without loss of generality, assume that $\lim_{n\to\infty} \frac{t_n-\tau_n^q}{t_n} = 0$. This implies that

$$\lim_{n\to\infty} \frac{W_{t_n}^q}{t_n} = \lim_{n\to\infty} \left[\frac{\Sigma^q(\tau_n^q, t_n)}{t_n} - \frac{\sum_{m=1}^{M} R_m^q(T_m(t_n) - T_m(\tau_n^q))}{t_n} \right] = 0,$$

which contradicts the assumption that $\lim_{n\to\infty} \frac{W_{t_n}^q}{t_n} = \eta^q > 0$. $\blacksquare$

The next lemma is a general result for inner products.

Lemma 3. *Let* $\vec{\eta}$, $\tilde{\eta} \in \mathbb{R}^Q$. *Suppose that* $\langle\vec{\eta}, \vec{\eta}\rangle_{\vec{\alpha}} < \langle\tilde{\eta}, \tilde{\eta}\rangle_{\vec{\alpha}}$, *then* $\langle\vec{\eta}, \tilde{\eta}\rangle_{\vec{\alpha}} < \langle\tilde{\eta}, \tilde{\eta}\rangle_{\vec{\alpha}}$.

Proof: The proof is a simple consequence of the fact that $\langle\vec{\eta} - \tilde{\eta}, \vec{\eta} - \tilde{\eta}\rangle_{\vec{\alpha}} > 0$. $\blacksquare$

Let $|| \cdot ||$ be the norm implied from the inner product, $\langle\cdot,\cdot\rangle_{\vec{\alpha}}$; that is, $||\vec{W}|| = \sqrt{\left\langle\vec{W}, \vec{W}\right\rangle_{\vec{\alpha}}}$. We are now in a position to prove Proposition 3.

Proof of Proposition 3: Let $\{t_n\}_{n=1}^{\infty}$ be an increasing, unbounded, sequence of time points such that $\liminf_{n\to\infty} \frac{t_n-t_{n-1}}{t_n} > 0$, and $\lim_{n\to\infty} \frac{||\vec{W}(t_n)||^2}{t_n^2} = \limsup_{t\to\infty} \frac{||\vec{W}(t)||^2}{t^2}$. Suppose, by contradiction, that, with positive probability, $\lim_{n\to\infty} \frac{||\vec{W}(t_n)||^2}{t_n^2} > 0$, and let $\{t_{n_k}\}_{k=1}^{\infty}$ be a subsequence of $\{t_n\}_{n=1}^{\infty}$, such that $\lim_{k\to\infty} \frac{\vec{W}(t_{n_k})}{t_{n_k}}$ exists, and is equal to a nonnegative, nonzero vector, $\vec{\eta}$. Without loss of generality, let $\{t_n\}_{n=1}^{\infty} = \{t_{n_k}\}_{k=1}^{\infty}$. The selection of $\{t_n\}_{n=1}^{\infty}$ and $\vec{\eta}$ implies, in particular, that

$$\langle \vec{\eta}, \vec{\eta} \rangle_{\vec{\alpha}} = \limsup_{t \to \infty} \frac{||\vec{W}(t)||^2}{t^2}. \tag{4}$$

For $n = 1, 2, ..$, let $\tau_n' = \sup \left\{ t < t_n \mid A(\vec{W}(t)) \not\subseteq A(\vec{\eta}) \right\}$ ($\tau_n' = 0$, if there is no such t), and define τ_n^q, for q such that $\eta^q > 0$ as in Claim 2. Also, let $s_n = \max \{t_{n-1}, \tau_n', \tau_n^q,\ q \textit{ such that } \eta^q > 0$
The definitions of these time sequences imply that:

1. $A(\vec{W}(t)) \subseteq A(\vec{\eta})$ for all $t \in (s_n, t_n]$.
2. If $\eta^q > 0$, then $W^q(t) > 0$ for all $t \in (s_n, t_n]$.
3. $\liminf_{n\to\infty} \frac{t_n - s_n}{t_n} = \epsilon > 0$ (without loss of generality, let $\lim_{n\to\infty} \frac{t_n - s_n}{t_n} = \epsilon$).

Hence, for q such that $\eta^q > 0$,

$$W^q(s_n) = W^q(t_n) - \Sigma^q(s_n, t_n) + \sum_{m \in A(\vec{\eta})} R_m^q (T_m(t_n) - T_m(s_n)) \tag{5}$$

Dividing through by s_n, and taking the limit over a converging subsequence (without loss of generality, assume that it is the whole sequence), we get, that for q such that $\eta^q > 0$,

$$\bar{\eta}^q := \lim_{n\to\infty} \frac{W^q(s_n)}{s_n} = \lim_{n\to\infty} \frac{t_n}{s_n} \frac{W^q(s_n)}{t_n} = \frac{1}{1-\epsilon} \left[\eta^q - \epsilon \left(\rho^q - \sum_{m \in A(\vec{\eta})} f_m R_m^q \right) \right] \tag{6}$$

Where $f_m \geq 0$ are independent of q, and $\sum_{m \in A(\vec{\eta})} f_m = 1$. Similarly, if q is such that $\eta^q = 0$, let

$$\bar{\eta}^q := \lim_{n\to\infty} \frac{W^q(s_n)}{s_n} \tag{7}$$

where s_n is perceived as a subsequence of the original sequence, for which these limits exist. Hence, the norm of $\vec{\eta}$ satisfies:

$$\begin{aligned} ||\vec{\eta}||^2 = \langle \vec{\eta}, \vec{\eta} \rangle_{\vec{\alpha}} &= \left\langle \vec{\eta}, (1-\epsilon)\bar{\eta} + \epsilon \left(\vec{\rho} - \sum_{m \in A(\vec{\eta})} f_m \vec{R}_m \right) \right\rangle_{\vec{\alpha}} \\ &= (1-\epsilon) \langle \vec{\eta}, \bar{\eta} \rangle_{\vec{\alpha}} + \epsilon \left\langle \vec{\eta}, \vec{\rho} - \sum_{m \in A(\vec{\eta})} f_m \vec{R}_m \right\rangle_{\vec{\alpha}} \end{aligned} \tag{8}$$

Lemma 2, applied to $\vec{W} = \vec{\eta}$, implies that,

$$0 < \langle \vec{\eta}, \vec{\eta} \rangle_{\vec{\alpha}} \leq (1-\epsilon) \langle \vec{\eta}, \bar{\eta} \rangle_{\vec{\alpha}} < \langle \vec{\eta}, \bar{\eta} \rangle_{\vec{\alpha}}. \tag{9}$$

Hence, from Lemma 3 we get,

$$\langle \vec{\eta}, \vec{\eta} \rangle_{\vec{\alpha}} < \langle \bar{\eta}, \bar{\eta} \rangle_{\vec{\alpha}} \tag{10}$$

which contradicts (4). ∎

4 Adaptive Batching

In some systems, switching from one service configuration to another takes a non negligible length of time (e.g. setup time), and hence policies that involve frequent configuration switchings might not be optimal. For such systems, it is desirable to design service policies that would minimize the number of switchings. In this section we propose a class of adaptive batching policies, that are rate stable, even in the presence of switching times. The main idea is that the scheduling of large batches of jobs is being planned in advance, so that each configuration is only used once, during the service epoch of any batch. This approach is particularly useful in systems that allow relatively long planning horizon, and do not require real time response to changes in the traffic loads.

In order to describe an adaptive batching policy, we first assume the existence of a *static* policy that is *asymptotically optimal*, and its scheduling time is *sub-linear* (examples of such policies can be found in [1] and [3]). A static policy is one that prescribes how to process an initial workload vector, when additional arrivals are not allowed into the system. Suppose that $\tilde{\pi}$ is a static processing policy, and let $e^{\tilde{\pi}}(\vec{W})$ be the emptying time associated with this policy (excluding switching times), when the initial workload vector is some $\vec{W} \in \mathbb{R}_{+}^{Q}$. The minimal emptying time, using a static policy is given by $\delta(\vec{W})$, and can be found by solving the linear program, $LP(\vec{W})$. Recall that $\vec{\Sigma}(s,t)$ is the total workload that arrives into the system in the time interval $(s,t]$, and that $\vec{\rho}$ is the vector of traffic intensities.

Definition. A static policy $\tilde{\pi}$ is *asymptotically optimal* if $\lim_{t\to\infty} \frac{e^{\tilde{\pi}}(\vec{\Sigma}(s,t))}{t-s} = \delta(\vec{\rho})$, almost surely, $\forall s$.

Suppose that there is some 'thinking' time, $T^{\tilde{\pi}}(\vec{W})$ involved in planning how $\tilde{\pi}$ would process the workload $\vec{W}$.

Definition. The scheduling time of a static policy $\tilde{\pi}$ is *sub-linear* if $\lim_{t\to\infty} \frac{T^{\tilde{\pi}}(\vec{\Sigma}(s,t))}{t-s} = 0$, almost surely, $\forall s$.

In addition, suppose that the policy $\tilde{\pi}$ is *non-idling.*

Definition. A static policy $\tilde{\pi}$ is *non-idling* if whenever the system is nonempty, either a switching from one configuration to another takes place, or at least one job is being processed.

We now describe an adaptive batching policy, π, based on a static policy $\tilde{\pi}$, that is asymptotically optimal, and has a sub-linear scheduling time. The batches are defined in an inductive manner. The first batch B_0 includes the jobs that are present in the system at time 0, and is empty if there are no such jobs. For n=1,2,.., let B_n be the set of jobs that arrive during the scheduling and processing time of the jobs in B_{n-1}. If there are no such jobs, then B_n includes the first job to arrive into the system, after the processing of the jobs in B_{n-1} has been completed. The scheduling of the jobs in B_n takes place immediately after this batch has been formed, and their processing starts as soon as the planning phase is done.

Proposition 4 states the rate stability of an adaptive batching policy that is based on a static policy that satisfies the abovementioned criteria. The proof is based on sample path arguments, and is omitted due to space considerations.

Proposition 4. *Suppose that the input flows $\{\mathbb{I}^q : q = 1, .., Q\}$ are rate ergodic, with traffic intensity vector $\vec{\rho} \in \mathbb{S}$ (that is, $\delta(\vec{\rho}) \leq 1$). Let π be an adaptive batching policy, that* *is based on an asymptotically optimal static policy that is also, non-idling, with sub-linear scheduling and total switching times. Then, the system is rate stable under π.*

5 Concluding Remarks

We have studied a general class of systems with interacting resources, through a unifying queueing model. For such systems, the Stability Region is identified as a distinct set of traffic intensities, and two families of policies are presented, that achieve this set as their stability region. Beyond stability, these policies also address very important practical considerations, such as: real time response, weighted class priorities, long planning horizon and setup cost avoidance.

The investigation of these policies is not fully exhausted yet. Many issues such as performance analysis, minimization of operating costs and quality of service guarantees, stronger forms of stability under more restrictive assumptions, are all interesting and challenging as directions for further research.

References

[1] Bambos, N. and Walrand, J. (1993) Scheduling and stability aspects of a general class of parallel processing systems. *Adv. Appl. Prob.* **25**, 176-202.

[2] Chen, H. and Mandelbaum, A. (1991) Discrete flow networks: bottleneck analysis and fluid approximations. *Mathematics of Operations Research* **16**, 408-446.

[3] Gans, N. and van Ryzin G. (1997) Optimal control of a multiclass, flexible queueing system. *Oper. Res.* **45(5)**, 677-693.

[4] Harrison, J. M. (1996) The BIGSTEP approach to flow management in stochastic processing networks. In F. Kelly, S. Zachary, and I. Ziedins, editors, *Stochastic Networks: Theory and Applications,* pages 57-90. Oxford University Press.

[5] Maglaras, C. (1998) Discrete-review policies for scheduling stochastic networks: fluid asymptotic optimality. *Annals of Applied Probability.* Submitted.

[6] McKeown, N., Anantharam, V. and Walrand, J. (1996) Achieving 100% throughput in an input-queued switch. *Proceedings of IEEE Infocom*, San Francisco.

[7] Tassiulas, L. and Ephermides, A. (1992) Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks. *IEEE Transactions on automatic control* vol. **37**, No. **12**, 1936-1948.

[8] Wasserman, K. M. and Lennon Olsen, T. (1998) On mutually interfering parallel servers subject to external disturbances. *Oper. Res.* Submitted.

A New Upper Bound on the Reliability Function of the Gaussian Channel

A. Ashikhmin [*]
Bell Labs, Lucent Technologies
600 Mountain Avenue
Murray Hill, NJ 07081
aea@research.bell-labs.com

A. Barg
Bell Labs, Lucent Technologies
600 Mountain Avenue
Murray Hill, NJ 07081
abarg@research.bell-labs.com

S. Litsyn [†]
Dept. of Electrical Engineering–Systems
Tel Aviv University
Ramat Aviv 69978, Israel
litsyn@eng.tau.ac.il

Abstract

We derive a new upper bound on the exponent of error probability of decoding for the best possible codes in the Gaussian channel. This bound is tighter than the known upper bounds (the sphere-packing and minimum-distance bounds proved in Shannon's classical 1959 paper and their low-rate improvement by Kabatiansky and Levenshtein). The proof is accomplished by studying asymptotic properties of codes on the n-dimensional sphere $S^{n-1}(\mathbb{R})$. First we prove that the distance distribution of codes of large size necessarily contains a large component. A general theorem establishing this estimate is proved simultaneously for codes on the Euclidean sphere and in real and complex projective spaces.

To derive specific estimates of the distance distribution, we study the asymptotic behavior of Jacobi polynomials $P_k^{ak,bk}$ as $k \to \infty$. This group of results provides the exact behavior of the exponent of Jacobi polynomials in the entire orthogonality segment.

Since on the average there are many code vectors in the vicinity of the transmitted vector $\mathbf{x}$, one can show that the probability of confusing $\mathbf{x}$ and one of these vectors cannot be too small. This proves a lower bound on the error probability of decoding and the upper bound announced in the title.

1 Overview of the background and results

The classical model of communication over channels with noise, introduced by Shannon in 1948, assumes that messages are represented by vectors (points) in the n-dimensional Euclidean space. Under this model it is assumed that when a vector $\mathbf{x}$ is sent over the channel, the received signal is represented by a vector $\mathbf{z} = \mathbf{x} + \mathbf{y}$, where $\mathbf{y}$ is a Gaussian vector whose coordinates are independent Gaussian variables with mean zero and variance σ^2.

---

[*] Research done while at Los Alamos National Laboratory, Los Alamos, NM 87545

[†] Research done while visiting DIMACS Center, Rutgers University, Piscataway, NJ 08854.

A consistent definition of capacity of such a channel is obtained if one assumes that the input signals satisfy some sort of energy constraints. Typically one assumes that the energy, or the average energy, of input signals does not exceed a given number $A\sigma^2$ per dimension, where A is a positive number called the "signal-to-noise ratio." Shannon [17] has shown that for a set of input signals of sufficiently large size the study of the channel is reduced to considering signals of *constant* energy equal to $\sigma\sqrt{An}$, that is, points on the n-dimensional sphere in $\mathbb{R}^n$. In this paper we restrict ourselves to this communication model, which will be referred to as the Gaussian channel (with discrete time and continuous amplitude). It suffices to consider spheres of any fixed radius. Therefore, in a large part of the paper we study only codes on the *unit* sphere in $\mathbb{R}^n$, denoted by $S^{n-1} = S^{n-1}(\mathbb{R})$. A *code* is a finite subset of $S^{n-1}(\mathbb{R})$. To distinguish between codes on S^{n-1} and in the Hamming space, the former are often called spherical codes.

Analogously to the Hamming case, the most important parameters of spherical codes studied in geometry and coding and information theory are the minimum distance and the error probability of decoding as functions of the code size. A natural geometric motivation for the distance problem is studying the best possible packings of $S^{n-1}(\mathbb{R})$ with spherical caps. This and closely related problems of finding the best possible fillings of $\mathbb{R}^n$ with identical spheres and the kissing number were studied long before the emergence of coding theory (see a survey in [7]). Spherical codes in information theory were introduced by Slepian [21] (paper based on a 1951 Bell Labs report) and Shannon [17]. However, studies in geometry and coding theory developed independently of each other until the second half of the 70-ies when important unifying steps were taken by Delsarte, Goethals, and Seidel [8], [9], and Kabatiansky and Levenshtein [10].

A. PARAMETERS OF SPHERICAL CODES. Let $C \in S^{n-1}$ be a code and $R(C) = \frac{1}{n}\ln|C|$ its rate, $0 \le R(C) < \infty$. The distance $\operatorname{dist}(\mathbf{x},\mathbf{y}) = \|\mathbf{x}-\mathbf{y}\|^{1/2}$ between two points in S^{n-1} can be also measured by the inner product $\langle\mathbf{x},\mathbf{y}\rangle = 1-\frac{1}{2}\operatorname{dist}^2(\mathbf{x},\mathbf{y})$ or by the geodesic distance on the sphere $\theta = \arccos\langle\mathbf{x},\mathbf{y}\rangle$. Each of these measures is convenient in some coding-theoretic problems. Accordingly, let

$$d(C) := \min_{\substack{\mathbf{x},\mathbf{y}\in C\\ \mathbf{x}\neq\mathbf{y}}} \operatorname{dist}(\mathbf{x},\mathbf{y}), \quad (0 \le d(C) \le 2),$$

$t(C) = \max\langle\mathbf{x},\mathbf{y}\rangle = 1-\frac{1}{2}d^2(C)$, and $\theta(C) = \arccos t(C)$.

Let

$$M(n,\theta) = \max_{C:\theta(C)\ge\theta} |C|$$

be the maximum size of a code on S^{n-1} of angular distance θ. Asymptotic properties of codes are characterized by the functions

$$\begin{aligned}\overline{R}(\theta) &= \limsup_{n\to\infty}\frac{1}{n}\ln M(n,\theta)\\ \underline{R}(\theta) &= \liminf_{n\to\infty}\frac{1}{n}\ln M(n,\theta).\end{aligned}$$

Usually one is interested in upper bounds on $\overline{R}(\theta)$ and lower bounds on $\underline{R}(\theta)$. Below we assume that these two sequences have a common limit and speak loosely of the maximum possible rate $R(\theta)$ of a code (sequence of codes C_n) of angular distance θ. Note that by [16], $R(\theta) = 0$ for $\pi/2 \le \theta \le \pi$.

Likewise define $d(R)$ to be the maximum distance of a code (sequence of codes) of rate $R(C_n) \ge R$.

Remark. Following [1] it is not difficult to prove that $\overline{R}(\theta)$ is a continuous function of θ. Indeed, by [20],[11] for any $\theta' > \theta$

$$(1-\cos\theta)^{(n-1)/2}M(n,\theta) \le \sqrt{2\pi n}(1-\cos\theta')^{(n-1)/2}M(n+1,\theta').$$

Applying this $t = \tau n$ times and putting (for simplicity) $\theta' = \theta + \tau$, we obtain

$$\overline{R}(\theta) \le \frac{1}{2}\ln\Big(\frac{1-\cos\theta'}{1-\cos\theta}\Big) + (1+\tau)\overline{R}(\theta').$$

Letting $\tau \to 0$, we see that $\overline{R}(\theta)$ is right semicontinuous. Putting $z = \theta'$ we see that $\overline{R}(z)$ is also left semicontinuous.

The best known lower bound on $R(\theta)$ is Shannon's sphere-packing bound [17]:

$$\begin{aligned} R(\theta) &\ge -\ln\sin\theta \quad (0 \le \theta \le \pi/2), && (1)\\ d^2(R) &\ge 2(1-\sqrt{1-e^{-2R}}) \quad (0 \le R < \infty), && (2) \end{aligned}$$

which relies on the same type of argument that the Varshamov-Gilbert bound for the Hamming space.

Best known upper bounds on $R(\theta)$ were derived in [10]. One of the main results of [10] states that

$$R(\theta) \le \frac{1+\sin\theta}{2\sin\theta}H\Big(\frac{1-\sin\theta}{1+\sin\theta}\Big) \quad (0 \le \theta \le \pi/2), \tag{3}$$

where $H(x) = -x\ln x - (1-x)\ln(1-x)$ is the entropy function. This bound admits a small asymptotic improvement for $0 < \theta < 63°$ [10]:

$$R(\theta) \le -(1/2)\ln(1-\cos\theta) - 0.099$$

(the real numbers here and below are approximate). There bounds can be also transformed to relate R and d (rather than θ). Indeed, let $\rho_{kl} = \rho_{kl}(R)$ be the root of the equation

$$R = (1+\rho)H\big(\frac{\rho}{1+\rho}\big), \tag{4}$$

$\rho_{kl} = (1/2)(1-\sin\theta)/\sin\theta$, $0 \le \rho_{kl} < \infty$. Then

$$\begin{aligned} d^2(R) &\le d^2_{kl}(R) := \frac{2(\sqrt{1+\rho_{kl}} - \sqrt{\rho_{kl}})^2}{1+2\rho_{kl}}, && (5)\\ d^2(R) &\le 2e^{-2R-0.137}. && (6) \end{aligned}$$

For $0.234 < R \le \infty$ bound (6) is better than (5).

B. Error Probability of Decoding. A systematic study of the error probability of decoding for spherical codes was initiated by Shannon in [17]. Let W be a code on the sphere of radius $\sigma\sqrt{An}$. We assume that code vectors for transmission are chosen with equal probability. Then

$$P_e(W) = \frac{1}{|W|}\sum_{\mathbf{x}\in W} P_e(\mathbf{x})$$

is the average error probability, where $P_e(\mathbf{x})$ is the error probability of decoding provided that the transmitted vector is $\mathbf{x}$. Let

$$D(\mathbf{x}, W) = \{\mathbf{z} \in S^{n-1} : \|\mathbf{z} - \mathbf{x}\| < \|\mathbf{z} - \mathbf{x}'\| \text{ for all } \mathbf{x}' \in W, \mathbf{x} \neq \mathbf{x}\} \qquad (7)$$

be the Voronoi region of $\mathbf{x}$ with respect to the code W. Then the optimal decoding rule, i.e., the one minimizing the average error probability, associates to the received vector $\mathbf{z}$ a code vector $\mathbf{x}$ such that $\mathbf{z} \in D(\mathbf{x}, W)$. (The definition ignores vectors $\mathbf{z}$ at the same distance from two or more code vectors since their probability is 0.) Under this decoding, the error probability $P_e(W)$ equals

$$P_e(W) = \frac{1}{|W|} \sum_{\mathbf{x} \in W} \Pr\{\mathbf{z} \in \cup_{\substack{\mathbf{y} \in W \\ \mathbf{y} \neq \mathbf{x}}} D(\mathbf{y}, W)\}, \qquad (8)$$

where the last probability equals the total probability, under the Gaussian distribution with mean at $\mathbf{x}$ and variance σ^2 along each coordinate, of the area of the sphere complementary to the decoding region of $\mathbf{x}$. Further, let

$$\begin{aligned} P_e(R, A, n) &= \min_{W : R(W) \geq R} P_e(W) \\ E(R, A, n) &= -\frac{1}{n} \ln P_e(R, A, n) \\ \overline{E}(R, A) &= \limsup_{n \to \infty} E(R, A, n) \\ \underline{E}(R, A) &= \liminf_{n \to \infty} E(R, A, n). \end{aligned}$$

Again we are interested in upper bounds on $\overline{E}(R, A)$ and lower bounds on $\underline{E}(R, A)$ as functions of R for given A. A common limit of these two functions, provided that it exists, is called the *reliability function* of the channel, denoted $E(R, A)$. By abuse of notation, below we speak of upper and lower bounds on $E(R, A)$.

Shannon [17] showed that $E(R, A) > 0$ for $R \in [0, C)$, where $C = (1/2)\ln(1 + A)$ is the capacity of the channel. In this interval $E(R, A)$ is bounded above by the sphere-packing bound [17]

$$E(R, A) \leq E_{sp}(\theta(R), A), \qquad (9)$$

where

$$E_{sp}(\theta, A) = \frac{A}{2} - \frac{\sqrt{A} g(\theta, A) \cos\theta}{2} - \ln(g(\theta, A) \sin\theta),$$

$g(\theta, A) = \frac{1}{2}(\sqrt{A}\cos\theta + \sqrt{A\cos^2\theta + 4})$, and $\theta(R) = \arcsin e^{-R}$ is the "sphere-packing" angle; cf. (1). Further, $E(R, A)$ is bounded below as follows [17]:

$$E(R, A) \geq \begin{cases} \frac{A}{4}(1 - \cos\theta(R)), & 0 \leq R \leq R_1, \quad \text{(I)} \\ \frac{A}{4}(1 - \cos\theta(R_1)) + R_1 - R, & R_1 \leq R \leq R_2, \quad \text{(II)} \\ E_{sp}(\theta(R), A), & R_2 \leq R \leq C, \quad \text{(III)} \end{cases} \qquad (10)$$

where $R_1 = \frac{1}{2}\ln\left(\frac{1}{2} + \frac{1}{2}\sqrt{1 + \frac{A^2}{4}}\right)$, $R_2 = \frac{1}{2}\ln\left(\frac{1}{2} + \frac{A}{4} + \frac{1}{2}\sqrt{1 + \frac{A^2}{4}}\right)$.

Bounds (9) and (10.III) show that $E(R, A)$ is known exactly for $R_2 \leq R \leq C$. Shannon [17] also proved the inequality

$$E(R, A) \leq \frac{A}{4}, \quad 0 \leq R \leq C, \qquad (11)$$

which implies that bound (10.I) is tight for $R = 0$. The proof of (11) in [17] used the minimum-distance argument (the probability of mixing two code vectors at a minimum distance) together with a Plotkin-type bound on the size of the code. Independently and earlier Plotkin-type bounds on $d(R)$ were proved by Rankin [16]. Later it was realized that one can abstract from the Plotkin bound and use Shannon's argument to establish a general minimum-distance bound on $E(R, A)$ (see [18] for discrete channels and [12] for the Gaussian channel). This bound has the form

$$E(R,A) \le E_{md}(R,A) = \frac{A}{8} d^2(R). \tag{12}$$

Together with (5) this implies the best known bound on $E(R, A)$ for low code rates.

Finally, as shown in [18], the reliability function of discrete channels is bounded above by the straight line connecting any point of the minimum-distance bound with any point of the sphere-packing bound. Sheverdyaev [19] extended this result to the Gaussian channel, showing that a segment of the common tangent to $E_{md}(R, A)$ and $E_{sp}(R, A)$ gives an upper bound on $E(R, A)$ (note that both E_{md} and E_{sp} are convex). Rather than writing out a cumbersome explicit expression for this bound, we simply denote it by $E_{st}(R, A)$.

Concluding, let us summarize the results on the upper bound on $E(R, A)$ known to date:

$$E(R,A) \le \begin{cases} E_{md}(R,A), & 0 \le R \le R', \qquad \text{(I)} \\ E_{st}(R,A), & R' \le R \le R'', \qquad \text{(II)} \\ E_{sp}(R,A), & R'' \le R \le C, \qquad \text{(III)} \end{cases} \tag{13}$$

where R', R'' are certain numbers which are easier to compute for each given A than to write out in general.

In our paper, following [21], [17], we assume that code vectors can be any points on S^{n-1}. In communication theory one also studies a restricted case of this problem, namely transmission over the Gaussian channel with codes whose vectors have coordinates equal to (binary or nonbinary) roots of unity. Then it is possible [14] to obtain upper bounds on codes better in a certain region of rates than the Kabatiansky-Levenshtein bounds. For lower existence bounds on the reliability function of the Gaussian channel with *binary* codes see, e.g., [15].

C. OUTLINE OF THE PAPER. The goal of this paper is to prove a new upper bound on $E(R, A)$ given by the following theorem.

Theorem 1 *The reliability function of the Gaussian channel with signal-to-noise ratio A satisfies the upper bound*

$$E(R,A) \le \min_{0 \le \rho \le \rho_{kl}} \max_{w,d} \Big[\min\big(A\frac{d^2}{8}, A\frac{w^2}{8} - \mathcal{L}(w,d,\rho)\big) \Big], \tag{14}$$

where R is a value of the code rate,

$$0 \le d \le \frac{\sqrt{2}\,(\sqrt{1+\rho_{kl}} - \sqrt{\rho_{kl}})}{\sqrt{1+2\rho_{kl}}}, \quad d \le w \le \frac{\sqrt{2}\,(\sqrt{1+\rho} - \sqrt{\rho})}{\sqrt{1+2\rho}},$$

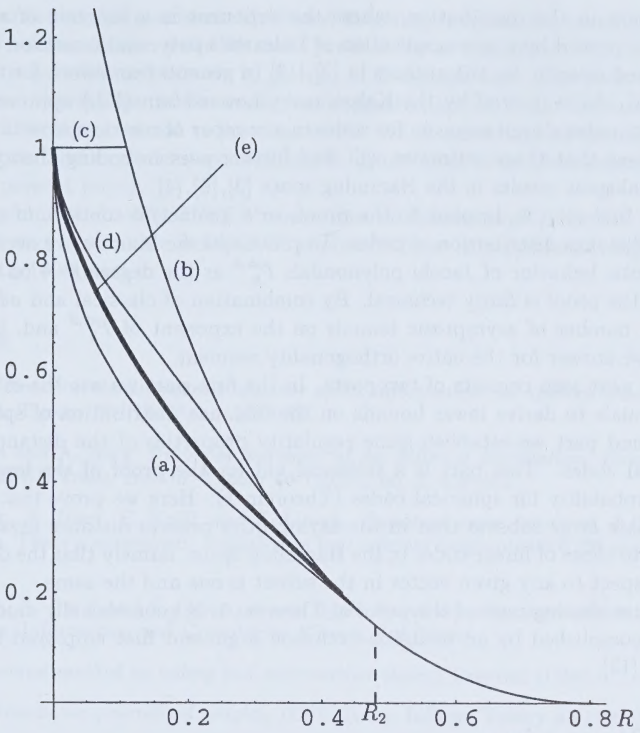


Figure 1: Bounds on the reliability function ($A = 4$): (a) "random coding" exponent (10.I-III), (b) sphere-packing bound (9), (c) minimum-distance bound (11), (d) minimum-distance bound (12), (e) the new bound (14). R_2 denotes the critical rate. Both curves (d) and (e) include a segment of the common tangent to the curve and the sphere-packing exponent.

ρ_{kl} *is the root of* $R = (1+\rho)H\left(\frac{\rho}{1+\rho}\right)$,

$$\mathcal{L}(w, d, \rho) = \min\left\{\frac{Ad^2w^2}{8(4w^2 - d^2)}, F\left(1 - \frac{1}{2}w^2, \rho\right)\right\},$$

$$F(x, \rho) = R - (1+\rho)H\left(\frac{\rho}{1+\rho}\right) + \int_x^1 \frac{4\rho(1+\rho)dz}{z + \sqrt{z^2 - 4(1-z^2)\rho(1+\rho)}}.$$

Together with a segment of a common tangent to the curve on the right-hand side of (14) and the sphere-packing exponent (9) this theorem improves bounds (13.I)-(13.II) for all rates $R \in (0, R'']$ (see Fig. 1).

The proof combines geometric, analytic, and combinatorial arguments, and is accomplished in several steps, each producing a result of its own. First we show that for sufficiently large n any code of large size has a rapidly growing component in the distance distribution. Without further explanation this is of course trivial. What we actually prove is that every segment $[d(C), d(C) + s]$ contains an exponentially growing component in the distribution, where the exponent is a function of s. This group of results is proved by a new application of Delsarte's polynomial method in coding theory, discovered recently by the authors in [2], [13] (a general framework for the method is set up in [5]. As suggested by the Kabatiansky-Levenshtein (KL) approach [10], we prove these estimates simultaneously for codes in a number of metric spaces including $S^{n-1}(\mathbb{R})$. We believe that these estimates will find further uses in coding theory as it happened with analogous results in the Hamming space [3], [6], [4].

The first step is devoted to the proof, in a geometric context, of general estimates on the distance distribution of codes. To prove specific bounds, we need to establish the asymptotic behavior of Jacobi polynomials $P_k^{\alpha,\beta}$ as the degree $k \to \infty$ and $\frac{\alpha}{k} \to a$. This part of the proof is fairly technical. By combination of classical and *ad hoc* methods we prove a number of asymptotic bounds on the exponent of $P_k^{\alpha,\beta}$ and, in a sense, give a definitive answer for the entire orthogonality segment.

The next step consists of two parts. In the first part we use the estimates of Jacobi polynomials to derive lower bounds on the distance distribution of spherical codes. In the second part we establish some regularity properties of the distance distribution of spherical codes. This part is a technical aid for the proof of the lower bound on the error probability for spherical codes (Theorem 1). Here we prove that in any code one can isolate large subsets that in the asymptotics possess distance invariance properties similar to those of linear codes in the Hamming space, namely that the distance spectrum with respect to any given vector in the subset is one and the same.

The remaining part of the proof of Theorem 1, is geometrically much more intuitive. It is accomplished by an inclusion-exclusion argument first employed for the Hamming case in [13].

2 Discussion

The polynomial ("linear programming") method in coding theory was defined in the founding works of Delsarte in 1972-73. Its applicability to bounding the size of codes (and designs) was extended by Delsarte, Goethals, and Seidel [8], [9] to include the spherical case. Kabatiansky and Levenshtein [10] developed a general approach, based on harmonic analysis of compact groups, to derive bounds on packings in a very broad class of homogeneous spaces. Our paper further extends the scope of the polynomial method. Though technically speaking, it is devoted to the proof of Theorem 1, on a more conceptual level it involves a large circle of ideas some of whose consequences are yet to be realized. A particular case of the linear programming method studied in coding theory hitherto relies upon the equality $|C| = \int dB(x)$, where $B(x)$ is the average distance distribution, and positivity conditions of the Q-transform of $B(x)$. However, many other functionals of primary interest to coding/information theory, notably, the error probability of decoding, can be written as, or are related to, linear forms of the distance coefficients. This enables one to study bounds on these quantities in the same fashion as bounds on the size of codes and shows that many information-theoretic problems have their natural place in the geometric context of coding theory. Curiously, this possibility has been overlooked for about 25 years until having been explored recently in [2], [13], where we studied the discrete case.

Applications in [2] include bounds on the undetected error exponent, which is directly expressible via the distance coefficients. The same holds true for error probability of decoding up to any radius t for which the spheres around code vectors are disjoint. For larger t, and in particular, for maximum likelihood decoding (t equals the covering radius of the code) it is not possible to write the error probability as a function of the distance distribution, but is possible to estimate it. A way of doing this that takes into account the entire distance distribution of the code, was suggested in [13] (cf. "the three-point lemma" above). This is in contrast to previous works which relied only on the minimum distance of the code, and explains improvements in upper bounds on error exponents in [2], [13], and the present paper.

This paper is submitted to the IEEE Transactions on Information Theory on Feb. 2, 1999. A complete text is available as DIMACS report 99-06 at http://dimacs.rutgers.edu /TechnicalReports/1999.html.

References

[1] M. Aaltonen, *Notes on the asymptotic behavior of the information rate of block codes*, IEEE Trans. Inform. Theory **30** (1984), no. 1, 84–85.

[2] A. Ashikhmin and A. Barg, *Binomial moments of the distance distribution: Bounds and applications*, IEEE Trans. Inform. Theory **45** (1999), no. 2, 438–452.

[3] A. Ashikhmin, A. Barg, E. Knill, and S. Litsyn, *Quantum error detection, I-II*, IEEE Trans. Inform. Theory, submitted. Available as lanl e-prints quant-ph/9906126 and quant-ph/99-6131.

[4] A. Ashikhmin, A. Barg, and S. Litsyn, *A new upper bound on codes decodable into size-2 lists*, Numbers, Information and Complexity, Kluwer Publishers.

[5] ———, *Polynomial method in coding and information theory*, Preprint (1999).

[6] ———, *New bounds on generalized weights*, IEEE Trans. Inform. Theory **45** (1999), no. 4, 1258–1263.

[7] J. H. Conway and N. J. A. Sloane, *Sphere packings, lattices and groups*, Springer-Verlag, New York-Berlin, 1988, With contributions by E. Bannai, J. Leech, S. P. Norton, A. M. Odlyzko, R. A. Parker, L. Queen and B. B. Venkov.

[8] P. Delsarte, J. M. Goethals, and J. J. Seidel, *Bounds for systems of lines and Jacobi polynomials*, Philips Res. Repts. **30** (1975), 91*–105*.

[9] ———, *Spherical codes and designs*, Geometriae Dedicata **6** (1977), 363–388.

[10] G. Kabatyansky and V. I. Levenshtein, *Bounds for packings on the sphere and in the space*, Problemy Peredachi Informatsii **14** (1978), no. 1, 3–25.

[11] V. I. Levenshtein, *The maximal density of filling an n-dimensinal Euclidean space with equal spheres*, Mat. Zametki **18** (1975), no. 2, 301–311, (In Russian. English translation Math. Notes 18 (1975), no. 1–2, 765–771 (1976).).

[12] ———, *Method of nonnegative definite functions in metric problems of coding theory*, Moscow, 1983, Doctor of Science Dissertation (In Russian).

[13] S. Litsyn, *New upper bounds on error exponents*, IEEE Trans. Inform. Theory **45** (1999), no. 2, 385–398.

[14] P. M. Piret, *Bounds for codes over the unit circle*, IEEE Trans. Inform. Theory **31** (1986), no. 6, 760–767.

[15] G. Sh. Poltyrev, *Bounds on the decoding error probability of binary linear codes via their spectra*, IEEE Trans. Inform. Theory **40** (1994), no. 4, 1284–1292.

[16] R. A. Rankin, *The closest packing of spherical caps in n dimensions*, Proc. Glasgow Math. Assoc. **2** (1955), 139–144.

[17] C. E. Shannon, *Probability of error for optimal codes in a Gaussian channel*, Bell Syst. Techn. Journ. **38** (1959), no. 3, 611–656.

[18] C. E. Shannon, R. G. Gallager, and E. R. Berlekamp, *Lower bounds to error probability for codes on discrete memoryless channels, II*, Information and Control **10** (1967), 522–552.

[19] A. Yu. Sheverdyaev, *Decoding methods in channels with noise*, Moscow, 1971, Ph. D. Thesis (in Russian).

[20] V. M. Sidelnikov, *New estimates for the closest packing of spheres in n-dimensional Euclidean space*, Mat. Sb. (N.S.) **95(137)** (1974), 148–158, 160.

[21] D. Slepian, *Group codes for the Gaussian channel*, Bell Syst. Techn. Journ. **47** (1968), 575–602.

Recursive decoding of Reed-Muller codes [*]

Ilya Dumer
Dept. of Electrical Engineering
University of California
Riverside, CA 92521
dumer@ee.ucr.edu

Abstract

New soft- and hard decision decoding algorithms are presented for general Reed-Muller codes $\left\{{m \atop r}\right\}$ of length 2^m and distance 2^{m-r}. We use Plotkin $(u, u+v)$ construction and decompose code $\left\{{m \atop r}\right\}$ onto subblocks $u \in \left\{{m-1 \atop r}\right\}$ and $v \in \left\{{m-1 \atop r-1}\right\}$. In decoding, we first try to find a subblock v from the better protected code and then proceed with the block u. The likelihoods of the received symbols are recalculated in a way similar to belief propagation. Thus, decoding is relegated to the two constituent codes. We repeat this recursion and execute decoding only at the end nodes $\left\{{j \atop 1}\right\}$ and $\left\{{j \atop j-1}\right\}$. The overall complexity has low order of $n \log n$. It is shown that this decoding substantially outperforms other algorithms of polynomial complexity known for RM codes. In particular, for medium and high code rates, the algorithm corrects most error patterns of weight $d \ln d/2$.

1 Introduction

The Reed-Muller code $RM(r,m)$ [7], which will be denoted below $\left\{{\mathrm{m} \atop \mathrm{r}}\right\}$, has parameters

$$n = 2^m,\ k = \sum_{i=0}^{r} \binom{m}{i},\ d = 2^{m-r}.$$

To construct this code, consider all polynomials $z(x_1, ..., x_m)$ of degree r or less taken over m Boolean variables. Then a codeword $\mathbf{z}$ is an ordered set of all 2^m values that polynomial z takes on these variables. It is also well known [7] that RM codes can be designed by repetitive employment of the Plotkin $(\mathbf{u}, \mathbf{u}+\mathbf{v})$ construction. Here the original block $(\mathbf{u}, \mathbf{u}+\mathbf{v}) \in \left\{{\mathrm{m} \atop \mathrm{r}}\right\}$ is represented by two subblocks $\mathbf{u} \in \left\{{\mathrm{m-1} \atop \mathrm{r}}\right\}$ and $\mathbf{v} \in \left\{{\mathrm{m-1} \atop \mathrm{r-1}}\right\}$. Now we can specify general $(\mathbf{u}, \mathbf{u}+\mathbf{v})$ construction as

$$\left\{{m \atop r}\right\} = \left\{{m-1 \atop r}\right\}, \left\{{m-1 \atop r}\right\} + \left\{{m-1 \atop r-1}\right\}, \tag{1}$$

---

[*] This research is supported by NSF grant No. NCR-9703844.

where the same codeword $\mathbf{u} \in \left\{ {m-1 \atop r} \right\}$ taken on both halves. In turn, we can split $\mathbf{u}$ and $\mathbf{v}$ further and obtain successively all codes $\left\{ {m \atop r} \right\}$. On step m, we can take the repetition code $\left\{ {m \atop 0} \right\}$ and full code $\left\{ {m \atop m} \right\}$, while all other codes $\left\{ {m \atop r} \right\}$ are obtained by recursion (1) from the previous step. Thus, any code $\left\{ {m \atop r} \right\}$ can be mapped onto the (m, r)-node of the Pascal triangle. Given any intermediate node (j, s), we move up and left to achieve the node $(j-1, s-1)$; or up and right to reach the node $(j-1, s)$. Note also that we can end our recursion on any level. In the sequel, we terminate our splitting at the nodes $(j, 1)$ (corresponding to biortogonal codes) or at the nodes $(j, j-1)$ (corresponding to the single parity check code). This is schematically shown in Fig. 1 for RM codes of the seventh order.

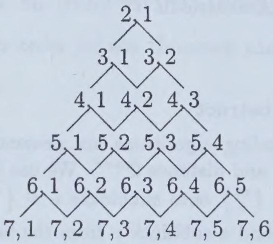


Figure 1: RM codes of length 128 on Pascal Triangle

Now let I_r^m denote the block of k information bits used to encode a vector $(\mathbf{u}, \mathbf{u}+\mathbf{v}) \in \left\{ {m \atop r} \right\}$. It is also important that recursion (1) splits I_r^m into two subblocks I_r^{m-1} and I_{r-1}^{m-1} that correspond to vectors $\mathbf{u}$ and $\mathbf{v}$, respectively. In this way, the new information strings are split again until we arrive at the end nodes $(j, 1)$ or $(j, j-1)$. Thus, any specific codeword can be encoded from (multiple) information strings assigned to the end nodes.

2 Background

Despite relatively bad code distance, RM codes have been considered in numerous publications thanks to efficient decoding procedures. *Majority decoding* developed in [9] (see also [2], [4], [8], and [10]) has complexity order at most nk and corrects all error patterns of weight below $d/2$. It is also known [5] that majority decoding corrects many error patterns of higher weights and can be further improved for soft-decision channels [13].

To decrease decoding complexity, *recursive algorithms* were also developed on both hard- ([6]) and soft-decision ([3]) channels. The algorithms provide for bounded distance decoding and have the lowest complexity order $n \min(r, m-r)$ known for RM codes to date. Simulation results presented in [12] also showed that recursive soft-decision algorithms can increase decoding domain of bounded distance decoding. Finally, efficient *permutation* algorithm considered in [11] for codes $\left\{ {m \atop 2} \right\}$, gives a slightly higher complexity $O(n^2 m)$ while correcting most error patterns of a higher weight $n(1-h)/2$, where h has a vanishing order of $(m/n)^{1/4}$ as $m \to \infty$.

In this paper, we wish to further develop recursive algorithms for RM codes and improve their performance both on short and long lengths. Such an improvement is especially important for short and moderate lengths on which RM codes are on par with the best codes known to date. Our goal is to use the same recursive presentation on Pascal triangle that is used above in code design. Namely, given an output $\mathbf{y} = (\mathbf{y}', \mathbf{y}'')$ with halves $\mathbf{y}'$ and $\mathbf{y}''$, we wish to perform two steps:

1. *combine* $\mathbf{y}'$ and $\mathbf{y}''$ to find $\mathbf{v} \in \left\{ {\mathrm{m-1} \atop \mathrm{r-1}} \right\}$.
2. *combine* $(\mathbf{y}', \mathbf{y}'')$ and $(\mathbf{0}, \mathbf{v})$ to find $\mathbf{u} \in \left\{ {\mathrm{m-1} \atop \mathrm{r}} \right\}$.

In turn, shorter codes $\left\{ {\mathrm{m-1} \atop \mathrm{r-1}} \right\}$ and $\left\{ {\mathrm{m-1} \atop \mathrm{r}} \right\}$ will be split further, while actual decoding procedures will be relegated to the end nodes. In the following sections, this procedure is discussed in more detail.

3 Decoding

Consider now the channel with Gaussian noise $\mathcal{N}(0, \sigma^2)$ and probability density function

$$G(y) = (1/\sqrt{2\pi}\sigma)e^{-y^2/2\sigma^2}. \tag{2}$$

The two symbols 0 and 1 are transmitted as +1 and −1. These two take arbitrary real values y at the receiver end with probability densities $G(y+1)$ and $G(y-1)$, respectively. In hard decision reception, we arrive at the BSC with transition error probability $p = Q(1/\sigma)$, where

$$Q(x) = \int_x^\infty e^{-y^2/2} dy/\sqrt{2\pi}.$$

In brief, we call these two channels AWGN(σ^2) and BSC(p) respectively.

Below we use the code $\left\{ {\mathrm{m} \atop \mathrm{r}} \right\}$ of length $n = 2^m$. Suppose that the codeword $\mathbf{z}$ is transmitted and $\mathbf{y} \in \mathcal{R}^n$ is received. Given any output signal $y \in \mathcal{R}$, we can find the posterior probabilities $p \overset{def}{=} p(1|y)$ and $q \overset{def}{=} p(0|y)$. By using the Bayes' rule we find

$$p = e^{-g/2}/(e^{g/2} + e^{-g/2}), \; q = e^{g/2}/(e^{g/2} + e^{-g/2}). \tag{3}$$

Here g is the *likelihood* of symbol 0 :

$$g = \log(q/p) = 2y/\sigma^2. \tag{4}$$

Finally, we introduce the *spread* h (which is the hyperbolic tangent of g) between the two probabilities q and p :

$$h = q - p = (e^{g/2} - e^{-g/2})/(e^{g/2} + e^{-g/2}) = \tanh(g). \tag{5}$$

Given an output vector $\mathbf{y} = (y_1, ..., y_n)$, we can find the quantities q_j, p_j, h_j, and g_j for any position j. In decoding, we will use the original vector $\mathbf{y}$, as well as the corresponding vectors $\mathbf{h} = (h_1, ..., h_n)$ and $\mathbf{g} = (g_1, ..., g_n)$. Note that $\tanh(g)$ is a one-to-one mapping. Therefore the three vectors are interchangeable:

$$\mathbf{y} \Leftrightarrow \mathbf{g} \Leftrightarrow \mathbf{h}. \tag{6}$$

We denote such a decoding $\mathbf{z} = \Psi_r^m(\mathbf{y}) = \Psi_r^m(\mathbf{h}) = \Psi_r^m(\mathbf{g})$, where $\mathbf{z} \in \left\{ {\mathrm{m} \atop \mathrm{r}} \right\}$ is our decoding result.

In ML decoding, we can first consider the string of hard decision outputs

$$a_j = \begin{cases} 0, & \text{if } y_j \geq 0, \\ 1, & \text{if } y_j < 0, \end{cases} \tag{7}$$

and try to find the most reliable codeword

$$\mathbf{z}^{*}: \sum_{j:z_j^{*} \neq a_j} |g_j| \leq \sum_{j:z_j \neq a_j} |g_j|$$

among all codewords $\mathbf{z} \in \left\{ {\mathbf{m} \atop \mathbf{r}} \right\}$. In our decoding below we also wish to minimize $\sum_{j:z_j \neq a_j} |g_j|$. However, this will be done on the premise that $\mathbf{y}$ is not heavily corrupted by noise. The corresponding threshold levels will be defined for BSC(p) and AWGN(σ^2) in Theorems 4 and 5, respectively.

3.1 Recalculating the probabilities

Let $\mathbf{z}'$ and $\mathbf{z}''$ denote the left- and right halves of any vector $\mathbf{z} \in \left\{ {\mathbf{m} \atop \mathbf{r}} \right\}$. We also use odd positions $j = 2s - 1$ on the left half and their even counterparts $2s$ on the right one for any $s = 1, ..., n/2$. Similar notations are used for all other vectors, say $\mathbf{y}, \mathbf{h}$ and $\mathbf{g}$. Given any vector $\mathbf{z} = (\mathbf{u}, \mathbf{u} + \mathbf{v})$ at the transmitter end, we can find

$$\mathbf{v} = \mathbf{z}' + \mathbf{z}'' (\operatorname{mod} 2) = (z_1 + z_2, z_3 + z_4, ..., z_{n-1} + z_n).$$

By contrast, at the receiver end we know only the strings $\mathbf{p} = (p_1, ..., p_n)$, $\mathbf{g}$, and $\mathbf{h}$ that define the probability distribution on the transmitted symbols. Our first problem is to find the spread $\mathbf{h}_*$ on vectors $\mathbf{v} = \mathbf{z}' + \mathbf{z}''$ given the original spread $\mathbf{h} = (\mathbf{h}', \mathbf{h}'')$ on vectors $\mathbf{z} = (\mathbf{z}', \mathbf{z}'')$.

Lemma 1 *(addition* mod 2*). Vectors* $\{\mathbf{z}' + \mathbf{z}''\}$ (mod 2) *have the spread*

$$\mathbf{h}_* = \mathbf{h}'\mathbf{h}'' = (h_1 h_2, ..., h_{n-1} h_n).$$

Given an output $\mathbf{y}$, we now find the probability spreads $\mathbf{h}$ and $\mathbf{h}^*$ on vectors $\mathbf{z}$ and $\mathbf{z}' + \mathbf{z}''$. Then our decoding $\Psi_r^m(\mathbf{h}) = (\mathbf{u}, \mathbf{u} + \mathbf{v})$ can first try to find $\mathbf{v} = \Psi_{r-1}^{m-1}(\mathbf{h}^*)$. Once vector $\mathbf{v}$ is found, we wish to find the remaining block $(\mathbf{u}, \mathbf{u} + \mathbf{v}) + (\mathbf{0}, \mathbf{v}) = (\mathbf{u}, \mathbf{u})$ in the second step of our decoding. Here we need to replace original symbols z_{2s} by symbols $z_{2s} + v_s$. Correspondingly, the latter have likelihoods

$$\overline{g}_{2s} = \begin{cases} g_{2s}, & \text{if } v_s = 0, \\ -g_{2s}, & \text{if } v_s = 1. \end{cases}$$

In other words, we change the sign of g_{2s} whenever $v_{2s} = 1$. The result is the string $\overline{\mathbf{g}}(\mathbf{u}, \mathbf{u}) = (\overline{\mathbf{g}}', \overline{\mathbf{g}}'')$, where the left half $\overline{\mathbf{g}}'$ is taken from the original vector $\mathbf{g}$ and equals $\mathbf{g}'$. The string $\overline{\mathbf{g}}(\mathbf{u}, \mathbf{u})$ represents likelihoods for arbitrary vectors $(\mathbf{u}, \mathbf{u})$ with two equal parts. Then we find the likelihoods $\mathbf{g}_*(\mathbf{u})$ given the two estimates $\overline{\mathbf{g}}', \overline{\mathbf{g}}''$.

Lemma 2 *(repetition). A string of likelihoods* $(\bar{\mathbf{g}}', \bar{\mathbf{g}}'')$ *defined on repeated vectors* $(\mathbf{u}, \mathbf{u})$ *gives for vectors* $\mathbf{u}$ *the string of likelihoods*

$$\mathbf{g}_*(\mathbf{u}) = \bar{\mathbf{g}}' + \bar{\mathbf{g}}''.$$

Once the likelihoods $\mathbf{g}_*$ are found, we execute decoding $\Psi^{m-1}_{r}(\mathbf{g}_*)$ that finds $\mathbf{u} \in \left\{ {m-1 \atop r} \right\}$. We perform both decodings $\Psi^{m-1}_{r-1}(\mathbf{h}_*)$ and $\Psi^{m-1}_{r}(\mathbf{g}_*)$ in a recursive way. In this process, we only recalculate spreads $\mathbf{h}$ and likelihoods $\mathbf{g}$ while using new (shorter) codes. Our recursion moves along the edges of Pascal triangle until recalculated vectors $\mathbf{h}$ and $\mathbf{g}$ arrive at the end nodes $(j, 1)$ and $(j, j-1)$. At the end nodes, we perform ML decoding. Now we can describe algorithm Ψ^{m}_{r} in a general soft-decision setting.

3.2 Recursive decoding for codes $\left\{ {m \atop r} \right\}$

1. Receive vector $\mathbf{y} \in \mathcal{R}^n$. Calculate $\mathbf{g} = \mathbf{g}^{m}_{r}$ and $\mathbf{h} = \mathbf{h}^{m}_{r}$ according to (4) and (5). *Call procedure* Ψ^{m}_{r}. Output decoded vector $\mathbf{z}^{m}_{r} \in \left\{ {m \atop r} \right\}$ and its information set I^{m}_{r}.

2. *Procedure* Ψ^{j}_{s}. Input $\mathbf{h}^{j}_{s}$ and $\mathbf{g}^{j}_{s}$.

2.1. Find $\mathbf{h}^{j-1}_{s-1} = (\mathbf{h}^{j}_{s})' \cdot (\mathbf{h}^{j}_{s})''$. Go to 3 if $s = 2$ or *call* Ψ^{j-1}_{s-1} otherwise.

2.2. Find $\mathbf{g}^{j-1}_{s} = (\bar{\mathbf{g}}^{j}_{s})' + (\bar{\mathbf{g}}^{j}_{s})''$. Go to 4 if $s = j-2$ or *call* Ψ^{j-1}_{s}otherwise.

2.3. Find $I^{j}_{s} = I^{j-1}_{s-1} \cup I^{j-1}_{s}$ and vector $\mathbf{z}^{j}_{s} = (\mathbf{z}^{j-1}_{s}, \mathbf{z}^{j-1}_{s} + \mathbf{z}^{j-1}_{s-1})$. *Return* vector $\mathbf{z}^{j}_{s} \in \left\{ {j \atop s} \right\}$ and its information set I^{j}_{s}.

3. Execute ML decoding Ψ^{j-1}_{1}. *Return* $\mathbf{z}^{j-1}_{1}$ and I^{j-1}_{1}.

4. Execute ML decoding Ψ^{j-1}_{j-2}. *Return* $\mathbf{z}^{j}_{j}$ and I^{j}_{j}.

Qualitative analysis. Note that increasing the noise power σ^2 reduces the means of the spreads $\mathbf{h}$ and likelihoods $\mathbf{g}$. In particular, it can be shown (see [13]) that for large noise $\sigma \to \infty$, the first two moments Eh and Eh^2 of the random variable $h = \tanh(2y/\sigma^2)$ satisfy the relation

$$Eh \sim Eh^2 \sim \sigma^{-2}. \tag{8}$$

In decoding process, we replace our original decoding Ψ^{m}_{r} by Ψ^{m-1}_{r-1}. The latter operates in a less reliable setting with the lower spread $\mathbf{h}^{m-1}_{r-1} = (\mathbf{h}^{m}_{r})' \cdot (\mathbf{h}^{m}_{r})''$ versus the original spread $\mathbf{h}^{m}_{r}$. Then the newly derived means $E(\mathbf{h}^{m-1}_{r-1})$ are the componentwise products of the corresponding means $E(\mathbf{h}^{m}_{r})'$ and $E(\mathbf{h}^{m}_{r})''$. According to (8), this multiplication is equivalent to replacing the original noise power σ^2 by the larger power σ^4. On the other hand, we also increase the relative distance d/n by using a better protected code $\left\{ {m-1 \atop r-1} \right\}$ instead of the original code $\left\{ {m \atop r} \right\}$. Our next decoding Ψ^{m-1}_{r} is performed in a better channel. Here we change the original likelihoods $\mathbf{g}^{m}_{r}$ for $\mathbf{g}^{m-1}_{r} = (\bar{\mathbf{g}}^{m}_{r})' + (\bar{\mathbf{g}}^{m}_{r})''$. As a result, our average likelihoods are doubled. This is equivalent to the map $\sigma^2 \Rightarrow \sigma^2/2$. In other words, code $\left\{ {m-1 \atop r} \right\}$ operates on a channel whose noise power is reduced two times. However, we also reduce the relative distance by using a weaker code $\left\{ {m-1 \atop r} \right\}$ instead of $\left\{ {m \atop r} \right\}$. In the next section, we present the *quantative* results of decoding performance.

4 Summary of results

We first consider hard decision reception on the channel BSC(p) with transition error probability $p = (1-h)/2$.

Theorem 3 *Recursive decoding of codes* $\left\{ {m \atop r} \right\}$ *used on a BSC(p), gives the output bit error probability* $\alpha \leq Q(\mu)$, *where*

$$\mu = 2^{(m-r)/2} h^{2^{r-1}} / \sqrt{1-h^{2^r}}, \quad p = (1-h)/2. \tag{9}$$

We then consider asymptotic capacity of recursive algorithms for long codes $\left\{ {m \atop r} \right\}$ as $m \to \infty$. We consider separately low-rate codes of fixed order r and those of fixed rate R. The latter implies that $r/m \to 0.5$. Let c be any constant exceeding $\ln 2$.

Theorem 4 *For* $m \to \infty$, *recursive decoding of codes* $\left\{ {m \atop r} \right\}$ *corrects most error patterns of weight:*

$$t \leq \begin{cases} n(1-(cm/d)^{1/2^r})/2, & \text{if } r = const, \\ d(\ln d - \ln 2m)/2, & \text{if } 0 < R < 1. \end{cases} \tag{10}$$

with decoding complexity order $n \log n$.

For fixed code rate R, the latter estimate is almost twice the bound $d \ln d/4$ known [5] for majority decoding. For low-rate codes of fixed order r we correct almost $n/2$ errors. In this case, our residual term $h = 1 - 2t/n$ has vanishing order of $(cm/d)^{1/2^r}$. The latter substantially reduces the former term $(m/d)^{1/2^{r+1}}$ known [5] for majority decoding. Such a performance was known before only for $r = 2$ and was obtained in permutation decoding presented in [11]. We note that this threshold h is now obtained for all orders r and is achieved with a lower complexity order of $n \log n$.

Our next issue is to improve recursive decoding by using soft decision likelihoods g. Given any output bit error rate $\alpha < 1/2$, we compare the corresponding noise powers σ_s^2 and σ_h^2 and transition error probabilities p_s and p_h that sustain this probability in soft and hard decision decoding.

Theorem 5 *Given any output bit error probability* α, *soft decision recursive decoding of long codes* $\left\{ {m \atop r} \right\}$ *of fixed order* r *increases* $\pi/2$ *times* α*-sustainable noise power of hard decision decoding:*

$$\sigma_s^2/\sigma_h^2 \to \pi/2, \quad m \to \infty. \tag{11}$$

Soft decision decoding of long codes $\left\{ {m \atop r} \right\}$ *of fixed code rate* R *increases* $4/\pi$ *times* α*-sustainable transition error probability of hard decision decoding:*

$$p_s/p_h \to 4/\pi, \quad m \to \infty. \tag{12}$$

The above theorem shows that for long low-rate RM codes we can gain $10\log_{10}(\pi/2) \approx 2.0$ dB over hard decision decoding for any output error rate α. The following corollary concerns the Euclidean weights of error patterns correctable by our algorithm. We show that for fixed r we exceed the bounded distance decoding weight $\sqrt{d}$ more than $2^{r/2}$ times. For fixed code rate R, we have a similar increase and outperform bounded distance decoding $2^{r/2}/\sqrt{m\ln 2}$ times.

Corollary 6 *For $m \to \infty$, soft decision recursive decoding of codes $\left\{ {m \atop r} \right\}$ corrects virtually all error patterns of Euclidean weight:*

$$\rho \leq \sqrt{n}(d/2m)^{1/2^r}, \qquad \textit{if } r = const, \tag{13}$$

$$\rho \leq \sqrt{n/m\ln 2}, \qquad \textit{if } 0 < R < 1. \tag{14}$$

5 Comparison

In Table 1 we compare the asymptotic performance of the newly developed algorithms with both the majority decoding and former recursive algorithms. This comparison is done for hard and soft decision decoding of low-rate codes of fixed order r and for codes of fixed rate R. As above, for low-rate codes we use the residual term h of our error-correcting capacity $n(1-h)/2$ for low-rate codes. Note that former recursive algorithms only provided for bounded distance capacity $d/2$. For soft decision decoding of low rate codes we use the squared Euclidean distance ρ^2. Again, the newly derived distance ρ^2 surpasses the one known for majority decoding. Finally, for medium code rates R we use the threshold weight t, for which decoding yet corrects most error patterns. In this case, we double decoding capacity of the former algorithms as seen from Table 1.

Table 1. Comparison of decoding algorithms for $\left\{ {m \atop r} \right\}$ codes.

Decoding capacity	Former Recursive	Majority Decoding	New Recursive
Hard decision, t for fixed r	$t = \frac{d}{2}$	$\frac{n}{2}\times(1-h)$, $h = (\frac{m}{n})^{1/2^{r+1}}$	$\frac{n}{2}\times(1-h)$, $h = (\frac{m}{n})^{1/2^r}$
Soft decision, ρ^2 for fixed r	$\rho^2 = \sqrt{d}$	$(\frac{n}{m})^{1/2^{r+1}}\sqrt{n}$	$(\frac{n}{m})^{1/2^r}\sqrt{n}$
Hard decision, t for fixed R	$t = \frac{d}{2}$	$d\ln d/4$	$d\ln d/2$

Below in Table 2, we present simulation results for bit error rates (BER) obtained by applying recursive and majority decoding to the code $\left\{ {9 \atop 4} \right\}$ of length 512. We also exhibit the results of computer simulation presented in [12]. Here, however, block error probabilities (BLER) were used in recursive decoding.

Finally, the last row represents a refined version of recursive decoding. This improvement uses the fact that recursive decoding gives different error rates at different end nodes. In particular, the worst error rates are obtained on the leftmost node $(m-r+1, 1)$, and the next worst results are obtained at the node $(m-r, 1)$. This asymmetrical performance can be justified by our qualitative analysis given above. We can see that the leftmost end node operates at the highest noise power σ^{2^r}.

The next important conclusion is to set the corresponding information bits as zeros. In this way we arrive at the subcodes of the original code $\left\{ {m \atop r} \right\}$ obtained by eliminating only a few least protected information bits. This "expurgation" procedure gives a substantial improvement to conventional recursive algorithms as seen from Table 2. Also, such a recursion gives good block error probabilities (BLER) in contrast to most iterative algorithms developed to date This part of the work performed jointly with K. Shabunov has been developed further and will be reported separately in more detail.

Table 2. Decoding performance for code $\left\{ {9 \atop 4} \right\}$.

SNR (dB)	2	3	4
Recursive [12]	0.9	0.5	0.2
Majority [13]	0.3	0.15	0.1
Recursive (new)	0.2	0.03	0.002
BER for subcodes	0.05	0.003	$3 \cdot 10^{-5}$
BLER for subcodes	0.2	0.02	$2 \cdot 10^{-4}$

6 Concluding remarks

It is interesting to compare the presented recursive algorithm with a few former variants considered in [3], [6], and [12]. Our algorithm is similar to these especially to the one presented in [3]. As a result, we achieve similar complexity of order $n \log n$. However, there are three differences. Instead of studying bounded-distance decoding, we try to find actual decoding capacity of recursive algorithms. Secondly, in this new setting we use probabilistic tools and explicitly recalculate posterior probabilities while moving along the edges of Pascal triangle. Finally, we use a different stopping rule and terminate any branch after reaching the codes $\left\{ {j \atop 1} \right\}$ of the first order. The algorithms considered before were terminated at codes $\left\{ {j \atop 0} \right\}$ of order zero. By using probabilistic tools described above, one can prove that using codes of order zero gives asymptotic performance similar to that of majority decoding. Therefore our increase in decoding capacity mostly results from a different stopping rule.

References

[1] R.G. Gallager, *Low-density Parity Check Codes.* Cambridge, MA: M.I.T. Press, 1963.

[2] W.C. Gore, "The equivalence of L-step generalization and a Reed decoding procedure," *IEEE Trans. Info. Theory,* vol. IT-15, pp. 184–186, 1969.

[3] G.A. Kabatyanskii, "On decoding of Reed-Muller codes in semicontinuous channels," *Proc. 2nd Int. Workshop "Algebr. and Combin. Coding Theory"*, Leningrad, USSR, 1990, pp. 87-91 (in Russian).

[4] V.D. Kolesnik, "Probabilistic decoding of majority codes," Probl. Info.Transmission, vol. 7, no. 3, pp. 193-200, 1971.

[5] R.E. Krichevskiy, “On the Number of Reed-Muller Code Correctable Errors,” *Dokl. Soviet Acad. Sciences,* vol. 191, pp. 541-547, 1970.

[6] S.N. Litsyn, “On decoding complexity of low-rate Reed-Muller codes,” *Proc. 9^{th} All-Union Conf. on Coding Theory and Info. Transmission,* Part 1, Odessa, USSR, pp. 202-204, 1988 (in Russian).

[7] F.J. MacWilliams and N.J.A. Sloane, *The Theory of Error-Correcting Codes,* North Holland, Amsterdam, 1977.

[8] J.L. Massey, *Threshold decoding.* Cambridge, MA: M.I.T. Press, 1963.

[9] I.S. Reed, “A class of multiple error correcting codes and the decoding scheme,” *IEEE Trans. Info. Theory,* vol. IT-4, pp.38-49, 1954.

[10] L.D. Rudolph, “Threshold decoding of cyclic codes,” *IEEE Trans. Info. Theory,* vol. IT-15, pp.590-592, 1969.

[11] V. Sidel'nikov and A. Pershakov, “Decoding of Reed-Muller Codes with a Large Number of Errors,” *Probl. Info. Transmission,* vol. 28, pp. 80-94, 1992.

[12] G. Schnabl and M. Bossert, “Soft-Decision Decoding of Reed-Muller Codes as Generalized Multiple Concatenated Codes,” *IEEE Trans. Info. Theory,* vol. 41, pp. 304-308, 1995.

[13] I. Dumer and R. Krichevskiy, “Soft Decision Majority Decoding of Reed-Muller Codes,” IEEE Trans. Info. Theory, vol. 45, no. 1, 2000, to appear.

LOSSLESS COMPRESSION IN CONSTRAINED CODING

John Fan (Stanford University)
Brian Marcus (IBM)[1]
Ron Roth (Technion)

Extended Abstract

1 Introduction

A *constrained system* (or simply a *constraint*) S is the set of all words that can be produced by a labeled finite directed graph. A word of length n is called an *n-block.* The *(d, k)-run-length-limited* (RLL) constraint is defined as the set of binary words whose 1's are separated by runs of zeros of lengths between d and k [4], [9], [10]. Such constaints are commonly used in magnetic and optical recording. In such applications, the "d" controls intersymbol interference and the "k" provides feedback for timing updates.

A main problem in constrained coding is the design of efficient, robust (to channel error), low-complexity encoders, which encode unconstrained (user) binary data sequences into sequences that satisfy a given constraint. The encoders operate at fixed rate $p : q$ (for some p, q), in a state-dependent fashion. Of course, all sequences produced by the encoder must obey the given constraint, and the encoder must be decodable. But, in addition, in order to control error propagation (after reception across a noisy channel), it is desirable for the decoder to be sliding-block, i.e., a q-block w is decoded to a p-block as a function of w as well as a bounded number, m, of past q-blocks and a bounded number, a, of future q-blocks. An encoder that has such a decoder is called a *rate $p : q$ (m, a)-sliding-block-decodable encoder.*

In *lossless data compression*, the roles of encoder and decoder are reversed: constrained sequences are compressed into unconstrained sequences where no distortion is permitted upon decompression. The duality between constrained coding and lossless data compression has been noted by several authors (see for example [8]), where data compression techniques such as arithmetic coding have been applied to constrained coding.

In this work, we transform a constrained coding construction (namely, a type of "reverse"-ACH algorithm [1], [7]) into a new lossless data compression construction. Here, constrained sequences are losslessly compressed into unconstrained sequences. In addition, in the noisy case (i.e., when the constrained system, viewed as a source, is noisy), it is desirable for the compressor to have limited error propagation.

---

[1] presented this work at Allerton

A *lossless compression code* consists of a *compressor* and an *expanding coder* (in short, *excoder*). The compressor is a sliding-block code from sequences of q-blocks of the constraint S to unconstrained binary sequences of p-blocks. The excoder, on the other hand, will have the form of a finite-state machine; of course, in order to recover all constrained sequences, the excoder must be capable of producing every sequence in S. Such an excoder will be called a *rate* $p:q$ (m,a)*-sliding block compressible excoder*. In the special case that both excoder and compressor are block codes, we call the excoder a *block excoder*.

We emphasize that the main distinction between an encoder and an excoder is the following:

- Image of E*n*coder $\subseteq$ Constraint
- Constraint $\subseteq$ Image of E*x*coder

2 Main Result

Given a constrained system S and desired rate $p:q$, how can one determine if there is a rate $p:q$ sliding block decodable encoder for S or a rate $p:q$ sliding block compressible excoder for S?

The capacity of S tells the answer (at least for a *finite memory constraint*, i.e., a constraint such that there is a presentation for which all sufficiently long words uniquely determine the terminal state; these include RLL constraints). Recall that the *capacity*, $C(S)$, of a constraint S is the asymptotic growth rate of the number of n-blocks that appear in S. The following result is well known.

Theorem 1: (Adler-Coppersmith-Hassner [1]): Let S be a finite memory constrained system with $p/q \leq C(S)$. Then there is a rate $p:q$ sliding-block decodable encoder for S.

In this work, we establish the analogous result for lossless compression codes:

Theorem 2: Let S be a finite memory constrained system with $p/q \geq C(S)$. Then there is a rate $p:q$ finite-state sliding-block compressible excoder for S.

The proofs and code constructions used in these results are based on state splitting. In Theorem 1, the sequence of state splittings is guided by a so-called *approximate eigenvector*. Theorem 2 involves a similar kind of vector, called a *super-vector*, which satisfies a reversed inequality. This leads to some interesting consequences. For instance, given a constraint and rate satisfying the capacity condition of Theorem 2, there is a unique smallest super-vector; but there is no analogous result for approximate eigenvectors.

3 Motivation

The new construction for lossless compression can be incorporated in a scheme for combining constrained coding and error-correcting coding (ECC), as suggested recently by Immink in [5]. In the *standard concatenation* scheme, information is first passed through an error-correction encoder and then a constrained encoder before being transmitted across the channel. At the receiver, the data is decoded via the constrained decoder and then the error-correction decoder. It makes sense for the constrained code to be the inner code, since sequences recorded on the channel must satisfy the given constraint.

Typically, the rates of constrained encoders have been close, but not *really* close, to the capacity of the constraint. The reason for this is that in order to approach capacity, long block lengths must be used. But since, in standard concatenation, decoding of the constrained code is performed before error correction, a small channel error may cause very large error propagation.

This motivated Bliss [2] and Mansuripur [6] to consider *modified concatenation*, in which the order of concatenation is reversed. In other words, the ECC becomes the inner code; thus, with very high probability, all channel errors will be corrected *before* the constrained decoder and thus there will be no errors to propagate. Of course, the ECC encoder may not produce sequences that obey the given constraint. However, if the ECC were a systematic code, then only the parity portion of the ECC codewords would violate the constraint. These portions are encoded separately by a conventional (lower rate) constrained encoder that has very little error propagation.

As stated, standard concatenation still retains one advantage: a channel error burst will be compressed (via the constrained decoder) before ECC decoding, thereby reducing the ECC overhead, compared to modified concatenation. Immink's solution ([5]) to this was to losslessly compress the constrained codeword before being input to the ECC encoder and to compress the channel outputs before being input to the ECC decoder; this reduces the ECC overhead in modified concatenation to almost the same level as in standard concatenation.

Examples of *block* compression codes for this purpose were given by Immink [5] and Fan and Calderbank [3]. Both of these papers served as inspiration for our work. The main difference between those papers and ours is that we allow the use of *sliding-block* compression codes.

4 An Example

Ultimately, the choice of a compressor-excoder pair involves a balance of the following four factors:

1. compression rate $p : q$;
2. edge effects (how many extra p-blocks are affected by the phasing of a burst);
3. effect of the sliding-block window length $\mathrm{m} + \mathrm{a} + 1$ (i.e., how many extra p-blocks may be affected by each error); and —
4. compatibility between the block length p and the symbol alphabet of the ECC.

Keeping these factors in mind, we examined four compression codes for the RLL $(2, \infty)$ constraint, whose capacity is approximately .5515:

1. a rate 4:6 ($\approx$ .6667) block excoder
2. a rate 8:13 ($\approx$ .6154) block excoder
3. a rate 4:7 ($\approx$ .5714) (0,1)-sliding-block compressible excoder
4. a rate 5:9 ($\approx$ .5556) (0,1)-sliding-block compressible excoder

The state splitting method used in the proof of Theorem 2, was used to construct specific realizations of 4:7 and 5:9 sliding-block compressible excoders. The former was realized by a 2-state excoder and the latter by a 10-state excoder. These excoders are no more complex than the rate 8:13 block excoder, and the 4:7 excoder may be much simpler.

For raw channel burst lengths L (in bits) between 1 and 300, we computed, for each code, the minimum redundancy (in bits) needed to correct the burst. We assumed a user bit stream of 4096 bits encoded into the RLL $(2, \infty)$ constraint. We also assumed single-error-correcting interleaved Reed-Solomon codes, optimized over ECC symbol size individually for each code and each burst length L.

For smaller burst lengths, the two block excoders perform better than the sliding-block compressible excoders because errors cannot propagate beyond a single block. But for larger burst lengths, the superior compression rates of the sliding block compressible excoders more than compensate for the edge effects caused by the sliding window. For burst lengths above $L = 80$, the 4:7 excoder beats or ties either of the block excoders. For burst lengths above 235, the 5:9 excoder beats or ties all of the others.

A complete proof of Theorem 2, as well as a more detailed discussion of specific codes and code comparisons, will appear in a paper by the authors, entitled *Lossless sliding-block compression of constrained systems*, in IEEE Transactions on Information Theory.

References

[1] R. Adler, D. Coppersmith, M. Hassner, *Algorithms for sliding block codes — an application of symbolic dynamics to information theory, IEEE Trans. Inform. Theory,* 29 (1983), 5–22.

[2] W.G. Bliss, *Circuitry for performing error correction calculations on baseband encoded data to elimijnate error propagation, IBM Tech. Discl. Bull.,* 23 (1981), 4633–4634.

[3] J.L. Fan, A.R. Calderbank, *A modified concatenated coding scheme, with applications to magnetic data storage, IEEE Trans. Inform. Theory,* 44 (1998), 1565–1574.

[4] K.A.S. Immink, *Coding Techniques for Digital Recorders,* Prentice Hall, New York, 1991.

[5] K.A.S. Immink, *A practical method for approaching the channel capacity of constrained channels, IEEE Trans. Inform. Theory,* 43 (1997), 1389–1399.

[6] M. Mansuripur, *Enumerative modulation coding with arbitrary constraints and post-modulation error correction coding and data storage systems, Proc. SPIE,* 1499 (1991), 72–86.

[7] B.H. Marcus, *Sofic systems and encoding data, IEEE Trans. Inform. Theory,* 31 (1985), 366–377.

[8] G.N.N. Martin, G.G. Langdon, S.J.P. Todd, *Arithmetic codes for constrained channels, IBM J. Res. Develop.,* 27 (1983), 94–106.

[9] B.H. Marcus, R.M. Roth, P.H. Siegel, *Constrained systems and coding for recording channels,* in *Handbook of Coding Theory,* V.S. Pless and W.C. Huffman (Eds.), Elsevier, Amsterdam, 1998, 1635–1764.

[10] B.H. Marcus, P.H. Siegel, J.K. Wolf, *Finite-state modulation codes for data storage, IEEE J. Sel. Areas Comm.,* 10 (1992), 5–37.

Modelling of Timed Discrete Event Systems

Rajinderjeet S. Minhas and W.M. Wonham
Department of Electrical and Computer Engineering
10 King's College Circle
University of Toronto
Toronto, Ontario, M5S 1G4, Canada
minhas@control.toronto.edu, wonham@control.toronto.edu
(Please direct all the correspondence to the first author)

Abstract

In this paper we present a compact model of timed discrete event systems. This model is constructed by utilizing a dynamic programming solution to the problem of finding a shortest and a longest path in a timed transition graph. The model is linguistically equivalent to a timed activity transition graph and additionally carries timing information. While the model is not guaranteed always to be more economical than the corresponding timed transition graph, for many systems it is indeed much more compact.

1 Introduction

Discrete event systems (DES) are systems or processes that are discrete in both time and state space. They are generally modelled by automata in one form or another, including finite state machines and Petri nets. A DES is assumed to change its state upon the occurrence of events; events may occur spontaneously and/or nondeterministically. A theory for the modelling and control of DES was initiated by Ramadge and Wonham [7]; we refer to this framework as RW. A number of researchers have extended RW by incorporating features like modular, decentralized and hierarchical modelling and control. For instance, in [11], [8], [6] the authors propose modular and decentralized control while in [12] and [13] the authors investigate hierarchical control. Brandin and Wonham further extended RW by incorporating the notion of time via the introduction of a special event *tick* [4],[3]. This is done by associating lower and upper time bounds that constrain the occurrence of events; the corresponding DES is called an activity transition graph (ATG). An ATG is then converted into a timed transition graph (TTG) where the occurrence of *tick* is intended to signal the passage of a unit of time. A DES with *tick*s is referred to as a timed discrete event system (TDES). In a TDES, one means of control is forcing: certain events are defined to be forcible and can be used to preempt *tick*. In effect, forcing represents the scenario where a supervisor causes an event to take place before a deadline. Thus due to the ability to force events, a TDES supervisor may be able to synthesize a larger controllable behaviour than a DES supervisor acting without timing information.

Recently, Wong and Wonham have proposed a framework for the hierarchical control of DES [9] that extends the work of Zhong and Wonham [13]. They investigate hierarchical control in terms of controllable languages, define a concept called *control consistency,* and show it to be equivalent to hierarchical consistency [13]. Wong and Wonham adapted their framework for TDES in [10]. They show that the class of TDES in the setting of [3] is not closed under the control proposed there. They propose an alternative definition of TDES and introduce a new timed control technology. Finally they derive conditions for control consistency for the hierarchical control of TDES. However, Wong and Wonham also explicitly model time using *tick*s. The additional expressiveness and larger controllable behaviour come at a price: increased computational complexity, as the introduction of ticks results in larger models. Even a modest increase in the size of individual plant models can result in a huge increase in size when we form their composition. This is because the size of the composed model increases multiplicatively with respect to the sizes of the component models. Therefore, it would be desirable, if possible, to model time without incurring the heavy overhead associated with *tick*s. In [2], Brandin proposed to model time without using *tick*s explicitly, but just the timer information associated with events. The resultant model is linguistically equivalent to a timed activity transition graph (TATG), i.e. to a TTG from which *tick*s have been projected out. In addition, the proposed model retains timing information sufficient for control purposes. While there is no guarantee that this approach to the modelling of TDES will always be more efficient than the explicit modelling of TDES with *tick*s, there is a large number of systems for which it is indeed more efficient.

In this paper we take a constructive approach to derive a *tick*-less model in the spirit of Brandin [2]. We give a solution to the problem of finding shortest and longest strings (paths) in a TTG. Although this solution is not unique, it has the following advantage. At the state corresponding to the shortest (longest) path, the timers associated with the various events have decremented by the least (most) amount possible. Thus knowledge of the timer matrices corresponding to the shortest and longest paths at any state allow us to predict the earliest and latest possible occurrence of enabled events at that state. We give a method for computing these timer matrices without having to compute the corresponding strings. So, given an ATG, we can compute the corresponding TATG directly.

The rest of the paper is organized as follows. In Section 2 we present some of the preliminaries and specify the notation. In Section 3 we present our solution to the problem of finding a shortest and a longest path in a TTG. Finally, in Section 4 we present a method to directly compute the timer matrices corresponding to our solution in Section 3 and give an example to compare our approach with that of Brandin [2].

2 Notation and Preliminaries

Let Σ represent the alphabet and A represent the activity set with the convention that a_0 is the initial activity. As usual, let Σ^+ represent the language consisting of all possible strings of finite (non-zero) length. Let $\epsilon \notin \Sigma$ be the empty string; let $\Sigma^* := \Sigma^+ \cup \{\epsilon\}$ be the language comprising all possible strings of finite (possibly zero) length. Let t represent a *tick* of a global clock and let $\Sigma_t := \Sigma \cup \{t\}$. Let $P_t : \Sigma_t^* \rightarrow \{t\}^*$ represent the standard projection operator that erases all occurrences of non-*tick* events and let $Q_t : \Sigma_t^* \rightarrow \Sigma^*$ represent the standard projection operator that erases all occurrences of *ticks*. For any $\sigma \in \Sigma$, let l_σ and u_σ represent the associated lower and upper time bounds respectively. Let $L \subseteq \Sigma_t^*$ be a language represented by a given TTG and let $L_{act} \subseteq \Sigma^*$ be the language represented by the corresponding TATG. For $s \in L$, let $\text{Enab}_L(s)$ be the set of enabled events at s in L; let $\text{Elig}_s(L) = \{\sigma \in \text{Enab}_L(s) \mid s\sigma \in \text{Pre}(L)\}$ be the set of eligible events at s in L. For any fixed $n \in \mathbf{N}$, let $\alpha_1, \ldots, \alpha_n \in \Sigma$ represent some activities (with the possibility that $\alpha_i = \alpha_j$ for $i \neq j$) such that $w = \alpha_1\alpha_2 \cdots \alpha_n$ is a string in L_{act}. Define $S_w := \{s \in \Sigma_t^*\Sigma_{act} \cap L \mid Q_t(s) = w\}$ to be the set of all strings in L that don't terminate with a *tick* and are equivalent under *tick*-projection. We shall often refer to the elements of $S_{\alpha_1 \cdots \alpha_n}$ as paths between ϵ and α_n. Assume $\Sigma = \{\sigma_1, \ldots, \sigma_m\}$ for some $m \in \mathbf{N}$ and let $s \in L$. Define

$$T_s := \begin{bmatrix} \bar{t}_1 & \bar{t}_2 & \cdots & \bar{t}_m \\ \underline{t}_1 & \underline{t}_2 & \cdots & \underline{t}_m \end{bmatrix}$$

to be the timer matrix corresponding to s, i.e. T_s displays the state of each timer (for each event label in the alphabet Σ) at that node in the TTG of L which corresponds to the string s.

Example 1 *Consider a plant G whose ATG and the corresponding TTG are shown in Figure 1. Assume that the following triples represent the time bounds associated with the events:* $(\alpha, 1, 2)$, $(\beta, 2, 4)$ *and* $(\gamma, 3, 4)$. *For $w = \alpha\beta\gamma$, we have*

$$S_w = \left\{ \begin{array}{c} t\alpha t\beta t^2\gamma, t\alpha t\beta t^3\gamma, t\alpha t^2\beta t\gamma, t\alpha t^2\beta t^2\gamma, t\alpha t^3\beta\gamma, t\alpha t^3\beta t\gamma, \\ t^2\alpha\beta t^3\gamma, t^2\alpha\beta t^4\gamma, t^2\alpha t\beta t^2\gamma, t^2\alpha t\beta t^3\gamma, t^2\alpha t^2\beta t\gamma, t^2\alpha t^2\beta t^2\gamma \end{array} \right\},$$

while for $s = t\alpha t\beta$, we have

$$T_s = \begin{bmatrix} 2 & 4 & 3 \\ 1 & 2 & 2 \end{bmatrix}.$$

□

Now we define some notation for posing the shortest and the longest path problem in a dynamic programming framework.

For $1 \leq j \leq k \leq n$, let $M_j[i_{k+1}, \ldots, i_n] \subseteq L$ be defined as

$$M_j\left[i_{k+1}, \ldots, i_n\right] := \left\{ \begin{array}{c} t^{i_j}\alpha_j \cdots t^{i_k}\alpha_k \mid (\exists t^{i_1}\alpha_1 \cdots t^{i_{j-1}}\alpha_{j-1} \in L) \\ t^{i_1}\alpha_1 \cdots t^{i_{j-1}}\alpha_{j-1} t^{i_j}\alpha_j \cdots t^{i_k}\alpha_k t^{i_{k+1}}\alpha_{k+1} \cdots t^{i_n}\alpha_n \in L \end{array} \right\}$$

where

$$M_j[\,] := \left\{ \begin{array}{c} t^{i_j}\alpha_j \cdots t^{i_n}\alpha_n \mid (\exists t^{i_1}\alpha_1 \cdots t^{i_{j-1}}\alpha_{j-1} \in L) \\ t^{i_1}\alpha_1 \cdots t^{i_{j-1}}\alpha_{j-1} t^{i_j}\alpha_j \cdots t^{i_n}\alpha_n \in L \end{array} \right\}$$

denotes the set for $k = n$.

Example 2 *Again consider the plant G shown in Figure 1. Assume $w = \alpha\beta\gamma$ in L_{act} is the string of interest. Then, for instance,* $M_3[\,] = \{\gamma, t\gamma, t^2\gamma, t^3\gamma, t^4\gamma\}$, $M_1[\,] = S_w$, *and,* $M_1[3, 2] = \emptyset$. □

Define $\#t : \Sigma_t^* \to \mathbf{N}$ such that $\#t(s) = |P_t(s)|$, for all $s \in \Sigma_t^*$, where $|\cdot|$ represents the length of a string. Clearly, for any string (or path) $s = t^{i_j}\alpha_j \cdots t^{i_k}\alpha_k$ we have $\#t\,(s) = \sum_{m=j}^{k} i_m$; we shall refer to $\#t(s)$ as the length of s. For $1 \leq j, k \leq n$, define

$$\begin{aligned} f_j\left(i_{k+1}, \ldots, i_n\right) &= \min_{s \in M_j[i_{k+1}, \ldots, i_n]} \#t(s), \text{and } f_j() = \min_{s \in M_j[\,]} \#t\,(s)\,, \\ g_j\left(i_{k+1}, \ldots, i_n\right) &= \max_{s \in M_k[i_{k+1}, \ldots, i_n]} \#t(s), \text{and } g_j() = \max_{s \in M_j[\,]} \#t\,(s)\,. \end{aligned}$$

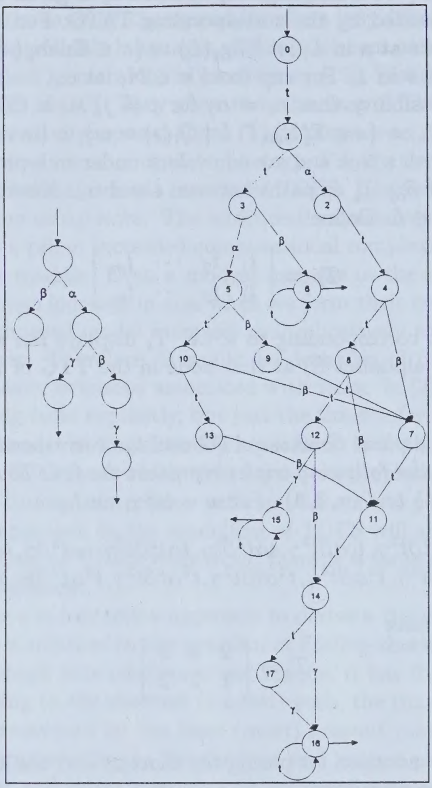

0
t
1
t
α
3
2
α
β
t
5
6
4
t
t
β
t
10
9
8
β
β
t
t
13
12
7
β
β
γ
t
15
β
11
t
t
14
t
17
γ
γ
18
t
β
α
β
γ
β
γ

Figure 1: ATG and TTG of G

In the above, by convention, if $M_j[i_{k+1},\ldots,i_n]=\emptyset$ then $f_j(i_{k+1},\ldots,i_n):=\infty$ and $g_j(i_{k+1},\ldots,i_n):=-\infty$.

The problem of finding the length of a shortest path of the form $t^{i_1}\alpha_1\cdots t^{i_n}\alpha_n$ in L may now be posed as the problem of evaluating $f_1()$. Similarly, the problem of finding the length of a longest path of the form $t^{i_1}\alpha_1\cdots t^{i_n}\alpha_n$ in L may be posed as the problem of evaluating $g_1()$.

For ease of notation, we define

$$\begin{array}{lll}
j_n^*=f_n(), & J_n^*=g_n(), & j'_{n-1}=f_{n-1}(j_n),\\
j_{n-1}^*=f_{n-1}(j_n^*), & J_{n-1}^*=g_{n-1}(J_n^*), & j'_{n-2}=f_{n-2}\left(j'_{n-1},j_n\right)\\
\vdots & \vdots & \vdots\\
j_1^*=f_1(j_2^*,\ldots,j_n^*), & J_1^*=g_1(J_2^*,\ldots,J_n^*), & j'_1=f_1\left(j'_2,\ldots,j'_{n-1},j_n\right).
\end{array}$$

3 Shortest and Longest Paths in a TTG

From the *Principle of Optimality* [1],[5] we know that past decisions should have no effect on an optimal decision. Thus the minimization problem may be represented [1],[5] as the recurrence relation

$$f_1() = \min_{\{j_n | t^{j_n}\alpha_n \in M_n[]\}} (j_n + f_1(j_n)) . \tag{1}$$

To compute $f_1()$, we have to find recursively the optimal paths in the sets of legal prefixes of the substrings of a given string. The complexity of such a solution is proportional to n^2 [5]. However, in a TTG, (1) can also be solved in a stepwise and non-recursive manner; the complexity of this solution is proportional to n.

To compute $f_1()$ we first find the shortest path between α_{n-1} and α_n in the TTG of L, i.e. we find the shortest string in $M_n[]$. We then successively find the shortest path to the preceding event in the set of its legal prefixes. So in a sense, to compute $f_1()$ we find the shortest path between successive events as we move backwards in the TTG of L. Therefore, the problem of finding the shortest overall path can be broken down into n intermediate problems of finding the shortest path to the previous event in the legal prefixes of a string. We begin by finding the shortest path between α_{n-1} and α_n i.e. we first evaluate j_n^*; then we find the shortest path from α_{n-2} to that point, i.e. we use the value of j_n^* to evaluate j_{n-1}^*; and so on until ϵ is reached i.e. until we can evaluate j_1^*. We call this the prefix solution to (1).

Theorem 3 *The shortest path problem*

$$f_1() = \min_{\{j_n | t^{j_n}\alpha_n \in M_n[]\}} (j_n + f_1(j_n)) \tag{2}$$

can also be solved by computing

$$\sum_{k=1}^{n} j_k^* . \tag{3}$$

Before attempting to find the maximal possible number of *ticks* in a path (in L) of the form $t^{i_1}\alpha_1 \cdots t^{i_n}\alpha_n$, we need to redefine the upper time limits associated with each event. The reason for this is illustrated by the following example. Let $n = 2$, i.e. consider $\alpha_1\alpha_2 \in L_{act}$. Assume $\mathrm{Enab}_{L_{act}}(\alpha_1) = \{\alpha_2, \sigma\}$ where $u_\sigma < u_{\alpha_2}$. Then if α_2 is to occur at all after α_1 then it must do so before u_σ *ticks*, otherwise it will be preempted by σ, which is forced to occur before u_σ *ticks*. We now generalize this reasoning.

Lemma 4 *For $\alpha_1 \cdots \alpha_n \in L_{act}$, the effective upper bounds for α_k, $2 \le k \le n$, are*

$$u'_{\alpha_k} = \min\{u_\sigma - t_\sigma \mid \sigma \in \mathrm{Enab}_{L_{act}}(\alpha_1 \cdots \alpha_{k-1})\},$$
$$u'_{\alpha_1} = \min\{u_\sigma \mid \sigma \in \mathrm{Enab}_{L_{act}}(\epsilon)\},$$

where σ has been enabled for t_σ ticks. We can assume that u'_{α_k}, $1 \le k \le n$, are finite because otherwise the longest time-path problem has a trivial solution: $g_n() = \infty$.

Theorem 5 *The longest path problem*

$$g_1() = \max_{\{j_n | t^{j_n}\alpha_n \in M_n[]\}} (j_n + g_1(j_n)) \tag{4}$$

can also be solved by computing

$$\sum_{k=1}^{n} J_k^*. \tag{5}$$

The prefix solution proposed in (3) and (5) is not unique. For instance, instead of moving backwards along a string, it is possible to move forward and find the piecewise shortest (longest) paths. Such a procedure also provides overall shortest and longest strings which may not be the same as the strings produced by the proposed method. We call this the postfix solution.

Example 6 *Let us again consider the timed transition graph of G as shown in Figure 1 and let $w = \alpha\beta\gamma$ be the string of interest. We can easily compute*

$$\begin{aligned}
j_3^* &= f_3() = \min_{s \in M_3[]} \#t(s) = 0, \\
j_2^* &= f_2(j_3^*) = f_2(0) = \min_{s \in M_2[0]} \#t(s) = 3, \\
j_1^* &= f_1(j_2^*, j_3^*) = f_2(3, 0) = \min_{s \in M_1[3,0]} \#t(s) = 1, \\
J_3^* &= g_3() = \max_{s \in M_3[]} \#t(s) = 4, \\
J_2^* &= g_2(J_3^*) = g_2(4) = \max_{s \in M_2[4]} \#t(s) = 0, \\
J_1^* &= g_1(J_2^*, J_3^*) = g_1(0, 4) = \max_{s \in M_1[0,4]} \#t(s) = 2.
\end{aligned}$$

Thus, the shortest path over S_w has $\min_{s \in S_w} \#t(s) = j_3^* + j_2^* + j_1^* = 0 + 3 + 1 = 4$ ticks and corresponds to $t\alpha t^3\beta\gamma$. Similarly, the longest path over S_w has $\max_{s \in S_w} \#t(s) = J_3^* + J_2^* + J_1^* = 4 + 0 + 2 = 6$ ticks and corresponds to $t^2\alpha\beta t^4\gamma$. However, as mentioned earlier, the prefix solution is not unique. Thus, for instance, the postfix solution yields $t\alpha t\beta t^2\gamma$ and $t^2\alpha t^2\beta t^2\gamma$ as the shortest and longest paths over S_w. □

4 Computation of TATG Directly from ATG

Let $s_1 = t^{j_1^*}\alpha_1 \cdots t^{j_n^*}\alpha_n$ and $s_2 = t^{J_1^*}\alpha_1 \cdots t^{J_n^*}\alpha_n$ be the prefix solutions for the shortest and longest paths over $S_{\alpha_1 \cdots \alpha_n}$; let $T_{s_1} = \begin{bmatrix} \overline{x}_1 & \cdots & \overline{x}_m \\ \underline{x}_1 & \cdots & \underline{x}_m \end{bmatrix}$ and $T_{s_2} = \begin{bmatrix} \overline{X}_1 & \cdots & \overline{X}_m \\ \underline{X}_1 & \cdots & \underline{X}_m \end{bmatrix}$ be the corresponding timer matrices. In addition, let $L = \begin{bmatrix} l_1 & \cdots & l_m \end{bmatrix}$ be a vector representing the relative order of enablement of events where $l_k \in \mathrm{N}$ for $1 \leq k \leq m$. So event σ_k is supposed to have been enabled after the occurrence of α_{n-l_k}. Now assume that $s_3 = s_5 t^{i_{n+1}^*}\alpha_{n+1}$ and $s_4 = s_6 t^{I_{n+1}^*}\alpha_{n+1}$ are the prefix solutions for the shortest and longest paths over $S_{\alpha_1 \cdots \alpha_n \alpha_{n+1}}$, where $s_5 = t^{i_1^*}\alpha_1 \cdots t^{i_n^*}\alpha_n$ and $s_6 = t^{I_1^*}\alpha_1 \cdots t^{I_n^*}\alpha_n$. Given T_{s_1}, T_{s_2} and L, we are interested in finding T_{s_3} and T_{s_4}. This can be done by first finding T_{s_5} and T_{s_6} and then appropriately adjusting the timer values by i_{n+1}^* and I_{n+1}^* *ticks* respectively.Without loss of generality, we can assume that $\alpha_{n+1} = \sigma_k$. Then we must have

$$i_{n+1}^* = \underline{X}_k, \text{ and } I_{n+1}^* = \min_{\{h \mid \sigma_h \in Enab_{L_{act}}(\alpha_1 \cdots \alpha_{n+1})\}} (\overline{x}_h).$$

Since s_1 is the shortest string over $S_{\alpha_1 \cdots \alpha_n}$ we get

$$\sum_{h=1}^{l_a} j_{n-h+1}^* \leq \sum_{h=1}^{l_a} i_{n-h+1}^* \tag{6}$$

for all $l_a > 0$. Similarly, since s_3 is the shortest string over $S_{\alpha_1 \cdots \alpha_{n+1}}$ we get

$$\sum_{h=1}^{l_a} i^*_{n-h+1} + i^*_{n+1} \leq \sum_{h=1}^{l_a} j^*_{n-h+1} + j^*_{n+1} \tag{7}$$

for all $l_a > 0$. Here j^*_{n+1} is the minimum number of *tick*s that must occur before α_{n+1} can occur from s_1. Combining (6) and (7) we get

$$\sum_{h=1}^{l_a} j^*_{n-h+1} \leq \sum_{h=1}^{l_a} i^*_{n-h+1} \leq \sum_{h=1}^{l_a} j^*_{n-h+1} + j^*_{n+1} - i^*_{n+1}. \tag{8}$$

We can easily compute T_{s_5} from (8) as we need only check the boundary conditions. A similar argument gives

$$\sum_{h=1}^{l_a} J^*_{n-h+1} \geq \sum_{h=1}^{l_a} I^*_{n-h+1} \geq \sum_{h=1}^{l_a} J^*_{n-h+1} + J^*_{n+1} - I^*_{n+1}. \tag{9}$$

Again, we can easily compute T_{s_6} from (9).

Example 7 *Consider the ATG given in Figure 2 and assume that the following triples define the timer values:* $(\alpha, 2, 7)$, $(\beta, 1, 5)$, *and* $(\gamma, 5, 6)$. *Further assume that we are only interested in strings of the form* $t^i \beta t^j \alpha t^k \gamma$. *Thus we will only need to compute a small subgraph of the TATG, i.e., the subgraph of the TATG corresponding to the string* $\beta\alpha\gamma$. *The TTG of this subgraph is also shown in Figure 3. The subgraph of the TATG computed using the proposed method is shown in Figure 2. As can be seen, the timer matrix corresponding to the shortest path of the form* $t^i \beta t^j \alpha t^k \gamma$ *is* $\begin{bmatrix} 7 & 5 & 6 \\ 2 & 1 & 5 \end{bmatrix}$. *From Figure 3 we see that this timer matrix corresponds to* $t^5 \beta \alpha \gamma$ *which is a shortest path. This is not a unique shortest path since* $t\beta t\alpha t^3\gamma$, $t\beta t^2\alpha t^2\gamma$, $t\beta t^3\alpha t\gamma$, $t\beta t^4\alpha\gamma$, $t^2\beta\alpha t^3\gamma$, $t^2\beta t\alpha t^2\gamma$, $t^2\beta t^2\alpha t\gamma$, $t^2\beta t^3\alpha\gamma$, $t^3\beta\alpha t^2\gamma$, $t^3\beta t\alpha t\gamma$, $t^3\beta t^2\alpha\gamma$, $t^4\beta\alpha t\gamma$, *and* $t^4\beta t\alpha\gamma$ *are also shortest paths.* □

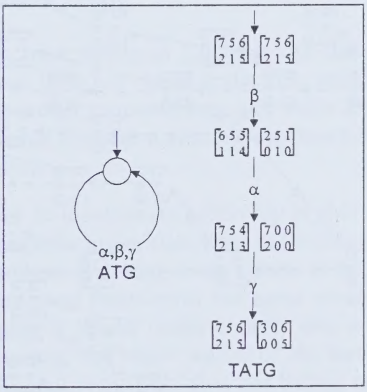


Figure 2:

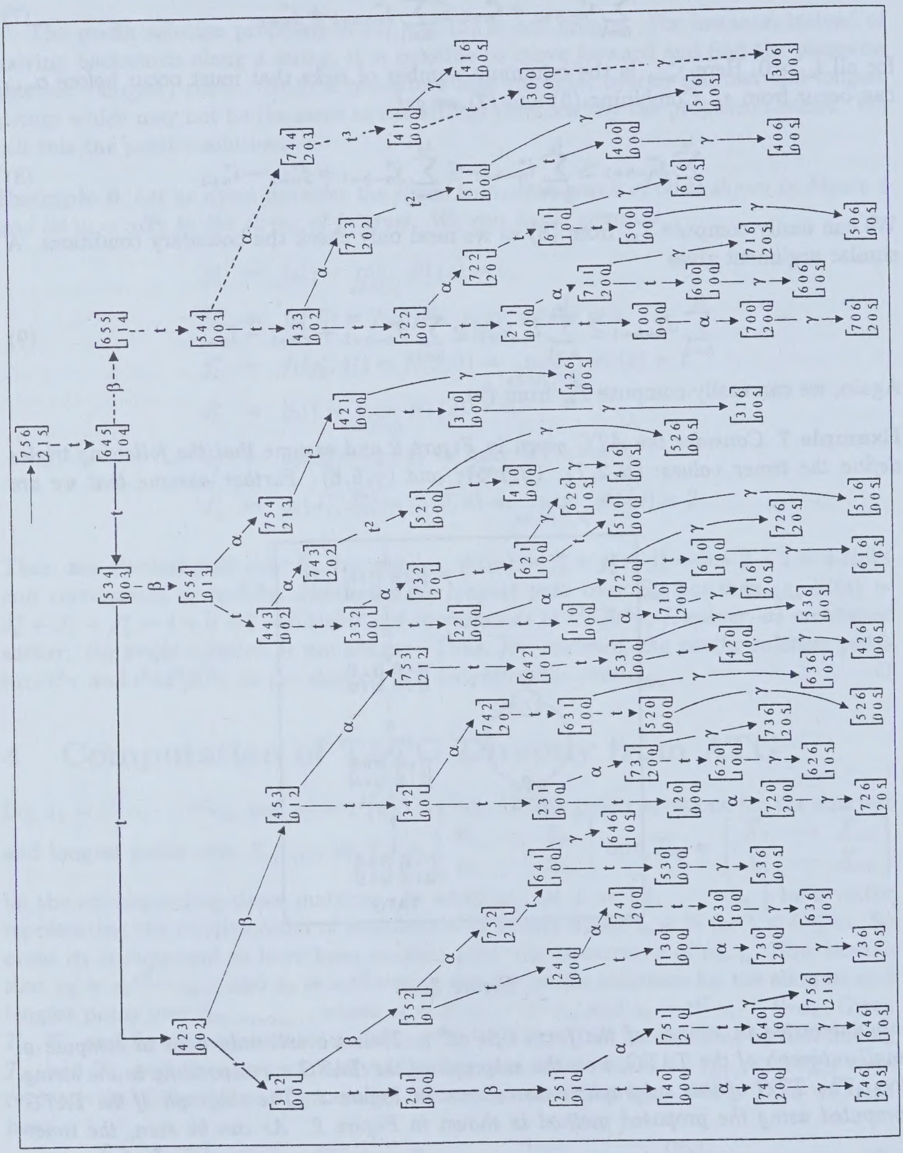


Figure 3:

Remark 1 *Using Brandin's method [2], the timer matrix corresponding to the shortest path of the form* $t^i\beta t^j\alpha t^k\gamma$ *is* $\begin{bmatrix} 7 & 2 & 6 \\ 2 & 0 & 5 \end{bmatrix}$. *This would imply that at most 2* ticks *may occur after the occurrence of a string of the form* $t^i\beta t^j\alpha t^k\gamma$ *before* β *is forced to occur. However, as can be seen from Figure 3, 5* ticks *may occur after* $t^5\beta\alpha\gamma$ *before* β *is forced to occur, i.e.* $t^5\beta\alpha\gamma t^5\beta$ *is a valid string in* L.

5 Conclusion and Work in Progress

In this paper, we have presented a *tick*-less model of timed discrete event systems. The model presented is inspired by a similar model proposed by Brandin in [2]. However, we have constructed our model by utilizing a dynamic programming solution to shortest and longest string (path) problem in a TTG. The solution is not unique but has the advantage that the timer matrices corresponding to the shortest and longest strings are sufficient for the purposes of modelling and control. Finally, we give a method for directly computing the timer matrices corresponding to the shortest and longest strings, i.e. without calculating the strings themselves. The results achieved by our approach differ from those obtained by Brandin's approach; the difference has been illustrated with an example.

Due to lack of space, a number of issues are not addressed in this paper; some of those issues are now discussed briefly.

- Wong and Wonham have shown in [10] that the class of TDES in the setting of Brandin and Wonham [3] is not closed under the control proposed there. Brandin's model in [2] and the model proposed here also suffer from the same problem. However, we have been able to make a suitable modification to ensure our model class is closed under control.

- It is not obvious how to incorporate forcibility in this *tick*-less model. Brandin [2] has proposed to change the upper time limit of forcible events to force them to occur before a desired deadline. But that doesn't seem to be a desirable solution because it associates different time limits with the same event label. The modification to our model that makes it closed under control also provides a way to implement forcing without changing the timer values of the forcible events. The idea is to incorporate an additional event that does not belong to the alphabet and use it for book-keeping.

- Since our model is linguistically equivalent to a TATG, and since a TATG is obtained from a TTG by projecting out the *tick* event, there is a possibility that the proposed model will have more states than a TTG. So while the proposed model may not always be more efficient than a TTG, there is a large number of systems for which it is indeed more compact. The TDES considered in Example 7 is one such system.

References

[1] Richard Bellman. *Adaptive Control Processes: a Guided Tour.* Princeton University Press, Princeton, NJ, 1961.

[2] Bertil A. Brandin. The modelling and supervisory control of timed discrete event systems. In *Proceedings of the International Workshop on Discrete Event Systems*, pages 8–13. IEE, August 1998.

[3] Bertil A. Brandin and W. Murray Wonham. Supervisory control of timed discrete-event systems. *IEEE Transactions on Automatic Control*, 39(2):329–342, 1994.

[4] Bertil Axel Brandin. *Real-Time Supervisory Control of Automated Manufacturing Systems.* PhD thesis, University of Toronto, 1993.

[5] Stuart E. Dreyfus. *Dynamic Programming and the Calculus of Variations.* Academic Press, New York, 1965.

[6] F. Lin and W. Murray Wonham. Decentralized supervisory control of discrete-event systems. *Information Sciences*, 44:199–224, 1988.

[7] P.J. Ramadge and W. Murray Wonham. Supervision of discrete event systems. In *Proceedings of IEEE, Special Issue on Discrete Event Dynamic Systems*, volume 37, pages 1692–1708. IEEE, 1982.

[8] Karen Rudie and W. Murray Wonham. Think globally, act locally: Decentralized supervisory control. *IEEE Transactions on Automatic Control*, 37(11):1692–1708, 1992.

[9] Kai C. Wong and W. Murray Wonham. Hierarchical control of discrete-event systems. *Discrete Event Dynamic Systems: Theory and Applications*, 6(3):241–273, July 1996.

[10] Kai C. Wong and W. Murray Wonham. Hierarchical control of timed discrete-event systems. *Discrete Event Dynamic Systems: Theory and Applications*, 6(3):275–306, July 1996.

[11] W. Murray Wonham and P. J. Ramadge. Modular supervisory control of discrete event systems. *Mathematics of Control, Signal & Systems*, 1(1):13–30, 1988.

[12] H. Zhong. *Hierarchical Control of Discrete-Event Systems.* PhD thesis, Department of Electrical Engineering, University of Toronto, 1992. Also appears as Technical Report 9208, Systems Control Group, Department of Electrical Engineering, University of Toronto.

[13] H. Zhong and W. Murray Wonham. On the consistency of hierarchical supervision in discrete-event systems. *IEEE Transactions on Automatic Control*, 35(10):1125–1134, October 1990.

Interacting Discrete Event Systems

Sherif Abdelwahed
sherif@control.toronto.edu

W. M. Wonham
wonham@control.toronto.edu

Department of Electrical and Computer Engineering
University of Toronto
Toronto, Ontario, Canada M5S 1A4

Abstract

In this paper we introduce interacting discrete event systems (IDES) as a modeling and analysis framework for complex discrete systems. An IDES is represented by a structure of two elements. The first is the set of components of the system and the second is a 'layout' structure that defines the way these components interact. We investigate formal properties of IDES. Also we show that under certain conditions on the layout of the system, verification and supervisory synthesis procedures can be decomposed into an equivalent set of procedures on the system components.

1 Introduction

Practical systems tend to be organized, in one way or another. An organized complex system can be identified as a set of interacting components. Each of these components performs a certain task and they interact together in a specified way. The relation between these components that defines how they communicate and interact together in the overall system is referred to as the systems architecture.

A discrete event system (DES) is a dynamical system with discrete time and state space, event-driven or asynchronous, and possibly nondeterministic. Examples include communication networks, database management systems, digital circuits and manufacturing systems. Complex discrete event systems are usually composed of a set of interacting components. In the least restrictive scenario, the components interact in a parallel scheme leading to exponential growth of the state space of the system. However, in general, parallel composition is not the default nor the desirable form of interaction. System components may interact in variety of ways to conform with their organizational constraints. In many practical situations, the components may have more than one type of interaction, usually arranged in a multi-level scheme. In order to deal with these systems efficiently, the modeling paradigm should be able to express and incorporate wide ranges of interactions.

Structured modeling techniques have emerged to provide a more faithful representation of complex systems by adding architectural information to the model. Architectural properties of a system can be utilized to decompose analysis procedures into a more manageable equivalent set of simpler procedures. In [5] Statecharts was proposed as a state-based language which extends conventional state machines with encapsulation and communication mechanisms. In a similar approach, high level Petri nets [3] were proposed to add a hierarchical decomposition mechanism to the conventional model. However, neither approach imposed any restriction on the model and hence few computational advantages, if any, are expected in these general settings. In the formal language field, there are some approaches that aim to provide a generalized form of parallel composition of languages. In [6] the operation of shuffle on trajectories was introduced as a generalization of the parallel composition of asynchronous systems.

Supervisory control theory (RW) for a general class of discrete event systems was initiated by Ramadge and Wonham [7]. Control-theoretic concepts such as controllability and observability have been introduced and formalized in the DES settings. Supervisory synthesis methods have been developed within this framework that guarantee that the controlled system behavior is contained in a given set of specifications. Within the RW framework there have been several attempts to incorporate architectural information directly into the model. In [1, 9] supervisory control theory was extended for a simplified version of statecharts. In [4] a linguistic approach to refinement based hierarchical control was investigated. Architectural properties in systems analysis and design have been also investigated under various DES frameworks [2, 8].

There is a need for a unified formulation that integrates the system architecture into the modeling and analysis procedures in a convenient and efficient way. In this paper we introduce a proposal that addresses these considerations. Practical systems are usually highly structured in spite of their apparent complexity. They are often organized in a multi-level hierarchy where the boundaries of the local modules are distinguishable. Moreover, the structure of these systems can be traced directly from the design and implementation process. Based on these observations, systems in our approach are represented by two elements; the first is a set of languages representing the components of the system and the second is a layout language that defines the way these components interact. This setup is referred to as one of *interacting discrete event systems* (IDES).

The paper is organized as follows. Section 2 introduces the basic elements of the multi-process environment referred to as a process space. The concepts of layout and generalized language composition are introduced in Section 3. In Section 4 we introduce the language based model for interacting discrete event systems and investigate decomposition of the verification problem for IDES. In Section 5 we investigate the decomposition of the supervisory control synthesis procedures under the IDES settings.

2 Elements of Process Spaces

Let Σ be an alphabet representing the events in a process under consideration. A string or word is sequence of events. We will write Σ^+ for the set of all nonempty finite strings with events in Σ, and $\Sigma^* = \Sigma^+ \cup \{\epsilon\}$, where $\epsilon \notin \Sigma$ denotes the empty string. A language over the alphabet Σ is any subset of Σ^*. A string $s' \in \Sigma^*$ is a prefix of $s \in \Sigma^*$, denoted $s' \leq s$, if there exists $u \in \Sigma^*$ such that $s'u = s$. The prefix closure of a language $H \subseteq \Sigma^*$, denoted $\overline{H}$, is the set of all strings in Σ^* that are prefixes of the strings in H. The function $\mathit{Last} : \Sigma^* \to \Sigma \cup \{\epsilon\}$ associates each string with its last event, that is $\mathit{Last}(a_1 a_2 \dots a_n) = a_n$ with the convention $\mathit{Last}(\epsilon) = \epsilon$.

We will extend the above notation to handle multi-process systems. Let I be the index set of a collection of processes. An alphabet vector over I is a set $\{\Sigma_i | i \in I\}$ of alphabets satisfying

$$(\forall i, j \in I) \quad \Sigma_i \not\subseteq \Sigma_j$$

This condition ensures that there is no redundancy in the alphabet set and autonomous action always exists. In the following we will use bold letters to distinguish vector quantities. Let $\mathbf{\Sigma} = \{\Sigma_i | i \in I\}$ be an alphabet vector. The union of all alphabets in $\mathbf{\Sigma}$ will be denoted $\alpha(\mathbf{\Sigma})$. A multi-process environment will be referred to as a process space. A process space is uniquely defined by its alphabet vector, hence both terms designate the same thing. We will write $\mathbf{\Sigma}^+$ for the set $\{\Sigma_i^+ | i \in I\}$ and $\mathbf{\Sigma}^*$ for $\{\Sigma_i^* | i \in I\}$. A language vector over $\mathbf{\Sigma}$ is a set $\mathbf{L} = \{L_i \subseteq \Sigma_i^* | i \in I\}$. The set of all language vectors over $\mathbf{\Sigma}$ is denoted $\mathbb{P}(\mathbf{\Sigma}^*)$. The language L_i is called the ith component of $\mathbf{L}$ and will be denoted $\mathsf{c}_i(\mathbf{L})$. Similarly a string vector is a set $\mathbf{s} = \{s_i \in \Sigma_i^* | i \in I\}$ and an event vector is a set $\boldsymbol{\sigma} = \{\sigma_i \in \Sigma_i | i \in I\}$. A language $L \subseteq \Sigma^*$ is considered a language vector $\mathbf{L}$ with a singleton element over the alphabet vector $\mathbf{\Sigma} = \{\Sigma\}$. For two language vectors $\mathbf{L}$ and $\mathbf{L}'$ over $\mathbf{\Sigma}$, we will say that $\mathbf{L}'$ is a subvector of $\mathbf{L}$ if for all $i \in I$ we have $\mathsf{c}_i(\mathbf{L}') \subseteq \mathsf{c}_i(\mathbf{L})$. In this case we write $\mathbf{L}' \sqsubseteq \mathbf{L}$. The componentwise union of $\mathbf{L}'$ and $\mathbf{L}$ is denote $\mathbf{L} \sqcup \mathbf{L}'$ and the componentwise intersection is denoted $\mathbf{L} \sqcap \mathbf{L}'$. Clearly, the set of language vectors over $\mathbf{\Sigma}$ together with the relation $\sqsubseteq$ form a complete lattice.

Let $\mathbf{\Sigma}$ be an alphabet vector with index I. The natural abstraction function over $\mathbf{\Sigma}$ is a function $f_{\mathbf{\Sigma}} : \alpha(\mathbf{\Sigma}) \to \mathcal{P}(I)$ defined as follows,

$$(\forall \sigma \in \alpha(\mathbf{\Sigma}))(\forall J \subseteq I) \quad f_{\mathbf{\Sigma}}(\sigma) = J \iff ((\forall i \in I)\, \sigma \in \Sigma_i \Leftrightarrow i \in J)$$

The $f_{\mathbf{\Sigma}}$ map abstracts labeling information from the set $\alpha(\mathbf{\Sigma})$ while preserving synchronization constraints imposed by the structure of $\mathbf{\Sigma}$. Basically, $f_{\mathbf{\Sigma}}$ associates with every event in $\alpha(\mathbf{\Sigma})$ the set of components in the process space $\mathbf{\Sigma}$ that have to coordinate in triggering this event. The kernel of the natural abstraction map satisfies

$$\ker f_{\mathbf{\Sigma}} = \{(\sigma, \sigma') \mid (\forall i \in I) \quad \sigma \in \Sigma_i \quad \Leftrightarrow \quad \sigma' \in \Sigma_i\}$$

In other words the relation $\ker f_{\mathbf{\Sigma}}$ partitions the events in $\mathbf{\Sigma}$ based on their synchronization requirements. The function $f_{\mathbf{\Sigma}}$ is extended to languages in the usual way. We will use the same symbol $f_{\mathbf{\Sigma}}$ for the extended function. The asynchronous subvector of $\mathbf{\Sigma}$ is defined as follows

$$\mathbf{\Sigma}_a = \{\Sigma_{ai} \mid (\forall \sigma \in \mathsf{c}_i(\mathbf{\Sigma}))\ \sigma \in \Sigma_{ai} \Leftrightarrow (\forall j \in I, j \neq i)\, \sigma \notin \mathsf{c}_j(\mathbf{\Sigma})\}$$

The set $\mathbf{\Sigma}_a$ is the maximal subvector of $\mathbf{\Sigma}$ with a disjoint set of alphabets. Finally, for the process space $\mathbf{\Sigma}$ we define the vector projection map $\mathbf{P}_{\mathbf{\Sigma}} : \alpha(\mathbf{\Sigma})^* \to \mathbb{P}(\mathbf{\Sigma}^*)$ which maps each language $L \in \alpha(\mathbf{\Sigma})^*$ to the language vector $\{P_i L | i \in I\}$ where $P_i : \alpha(\mathbf{\Sigma})^* \to \mathsf{c}_i(\mathbf{\Sigma})^*$ is the natural projection operation that erases all occurrences of $\sigma \in (\alpha(\mathbf{\Sigma}) - \mathsf{c}_i(\mathbf{\Sigma}))$. We may drop the script in $\mathbf{P}_{\mathbf{\Sigma}}$ if it is known from the context.

3 Generalized Language Composition

Let $\mathbf{\Sigma}$ be an alphabet vector. A layout over $\mathbf{\Sigma}$ is a language $K \subseteq \alpha(\mathbf{\Sigma})^*$ that is invariant with respect to the function $f_{\mathbf{\Sigma}}$, that is, $K = f_{\mathbf{\Sigma}}^{-1}(f_{\mathbf{\Sigma}}(K))$. K must satisfy

$$(\forall (\sigma, \sigma') \in \ker f_{\mathbf{\Sigma}})(\forall u, v \in \alpha(\mathbf{\Sigma})^*) \quad u\sigma v \in K \iff u\sigma' v \in K$$

We write $\mathsf{Y}(\Sigma)$ for the set of layouts over Σ. The vector projection of K, $\boldsymbol{P}K$, is called the domain of K. Clearly, this domain inherits the invariance with respect to f_Σ. A layout K is said to be asynchronous if the components of its domain have disjoint alphabets, otherwise it is a synchronous layout.

Proposition 1 The set $\mathsf{Y}(\Sigma)$ is closed under union, intersection, catenation and projection of cosets of f_Σ. □

The fact that the set of layouts corresponds directly to the set of languages over $\mathcal{P}(I)$ makes it closed under any linguistic operation that respects the natural abstraction function. The set of layouts is preserved in the expansion of the process space as stated in the following proposition.

Proposition 2 Let Σ and Π be two alphabet vectors; then

$$\Sigma \subseteq \Pi \implies \mathsf{Y}(\Sigma) \subseteq \mathsf{Y}(\Pi)$$

□

In the multi-process environment, layouts represent a unique form of abstract behaviour that distinguishes only the system components and their associated synchronization constraints as defined by the structure of the process space. Layouts can be used to constrain the components to follow a certain behavioral pattern. This can be achieved by synchronizing the behavior of the system components with the layout. This procedure will be referred to as the generalized composition operation.

Let K be a layout over Σ and $\boldsymbol{L} \sqsubseteq \Sigma^*$ be a language vector. The K-composition of $\boldsymbol{L}$, denoted $\|_K \boldsymbol{L}$, is the restriction of the synchronous product of the components of $\boldsymbol{L}$ to K. Formally

$$\|_K \boldsymbol{L} = K \cap \| \boldsymbol{L} = K \cap \bigcap_{i \in I} P_i^{-1} \mathsf{c}_i(\boldsymbol{L})$$

where $\| \boldsymbol{L}$ denotes the synchronous product of the components of $\boldsymbol{L}$. This composition scheme is the generalized composition operation.

Example 1 Let $\Sigma = \{(a, b, x), (c, d, x)\}$ be a process space. The associated abstraction function has a kernel $\ker f_\Sigma = \{(a, b), (x), (c, d)\}$. Will write u, x and v to denote the elements of $\ker f_\Sigma$ respectively. Let $\boldsymbol{L} = \{a(b + x), (c + x)d\}$ be a language vector. Let $K_1 = u^*v^*$, $K_2 = u^*xv^*$ and $K_3 = (uv)^*(u + \epsilon)$ be three different layouts over Σ. Then $\|_{K_1}\boldsymbol{L} = abcd$, $\|_{K_2}\boldsymbol{L} = axd$ and $\|_{K_3}\boldsymbol{L} = acbd$.

The general composition operation generalizes many standard linguistic operations. The following are examples of the layout of some standard binary operations (other asynchronous operations can be found in [6]). The relevant binary operations are defined over the alphabet vector $\Sigma = \{\Sigma, \Sigma'\}$; to simplify notation let $u = \Sigma - \Sigma'$, $v = \Sigma' - \Sigma$ and $x = \Sigma \cap \Sigma'$.

- Catenation is represented by the asynchronous layout $K = u^*v^*$. The domain of catenation is $\boldsymbol{P}K = \{u^*, v^*\}$. A more general synchronous catenation operation can be defined with layout $K = u^*x^*v^*$. The domain of synchronous catenation is $\{u^*x^*, x^*v^*\}$. In this form of catenation, one process starts, then may synchronize with a second process (if possible) and then the second process continues and exits.

- Synchronous product corresponds to the least restrictive layout, namely $K = \{u, x, v\}^*$. When $x = \emptyset$ the layout becomes that of the shuffle (asynchronous product) operation. Intersection can be obtained by the layout $K = x^*$. The domain in this case is $PK = \{x^*, x^*\}$.
- Refinement refers to the operation of replacing certain events in a language by new languages. Refinement takes different forms depending on the system. For example, in one setting an event is used to initiate the extension and another event is used to terminate it. This setting is usually referred to as handshaking. The layout of this form of refinement is given by $K = \{u, (xv^*x)\}^*$. In another setting only one event is used to initiate the extension. This interrupt-like refinement can be represented by a layout $K = \{u, (xv^*)\}^*$.
- Interleaving is another common form of interactions in software and hardware systems. In this form of interaction two systems (or more) are executed interchangeably based on certain time limit and priority rules. In the simplest setting interleaving can be represented by a layout $K = (v + \epsilon)(uv)^*(u^* + v^*)$. The domain of this interaction is $PK = \{u^*, v^*\}$. This corresponds to the case of two systems with equal priority, no time constraints and no shared events.

4 Algebraic Properties of Layouts

Layouts can be viewed as a class of abstract system dynamics which is, to certain extent, independent of the dynamics of the system components. Hence, algebraic properties of the layout language can provide a characterization of system behaviors. Owing to the abstract nature of layouts, this characterization does not depend on the process space but rather on the properties of the layout domain.

Completeness

A layout K over Σ is said to be complete if the K-composition of any language vector within the domain of K is not empty unless the synchronous product (free interaction) of its components is empty. Formally,

$$(\forall \boldsymbol{L} \sqsubseteq \boldsymbol{P}K) \quad \|_K \boldsymbol{L} = \emptyset \implies \|\boldsymbol{L} = \emptyset$$

Roughly speaking, a complete layout does not block any set of composable components within its domain.

Proposition 3 A layout $K \in \Upsilon(\Sigma)$ is complete if

$$(\forall \Sigma_{ai} \in \Sigma_a)(\forall \sigma \in \Sigma_{ai})(\forall u, v \in \alpha(\Sigma)^*) \quad u\sigma v \in K \implies u\Sigma_{ai}^* v \in K$$

□

On the base of the above proposition it is easy to check that the layouts of catenation, synchronous product, shuffle and refinements are all complete. The sufficient condition in the above proposition can be used to efficiently construct the minimal (in terms of the number of states and transitions) complete layout that contains a given layout.

Determinism

A layout $K \in \mathsf{Y}(\Sigma)$ is said to be deterministic if the K-composition of any string vector $\boldsymbol{s}$ over Σ is either a string or the empty set. Determinism reflects unambiguity in the system structure; each string in the global system behavior corresponds to a unique set of strings in components of the system. It is easy to check that catenation and refinement are deterministic while synchronous product and shuffle are not. One of the important features of deterministic layouts is that any language within that layout is decomposable into a unique set of components and that this decomposition is obtained by simple projection. This is stated as follows.

Proposition 4 A layout K is deterministic if and only if

$$(\forall H \subseteq K) \qquad \|_K \boldsymbol{P} H = H$$

□

It is straightforward to check that for an alphabet vector Σ the set of deterministic layouts in $\mathsf{Y}(\Sigma)$ is closed under catenation.

5 Interacting Discrete Event Systems

Let Σ be an alphabet vector with index set I. An interacting discrete event system over Σ is a tuple $\mathcal{L} = (\boldsymbol{L}, K)$ where $\boldsymbol{L} \sqsubseteq \Sigma^*$ is a language vector and $K \in \mathsf{Y}(\Sigma)$ is a layout. We will use calligraphic letters to denote IDES. With a slight abuse of notation we will write $\mathsf{c}_i(\mathcal{L})$ to denote the ith component of $\boldsymbol{L}$ in $\mathcal{L}$. The language generated by $\mathcal{L}$ is $L(\mathcal{L}) = \|_K \boldsymbol{L}$. Two IDES over some alphabet vector Σ are said to be matched if they have the same layout. The set of IDES over Σ with layout K will be denoted $\mathsf{M}(\Sigma, K)$. An IDES $\mathcal{L} = (\boldsymbol{L}, K)$ over Σ is said to be compact if

$$(\forall \mathcal{L}' \in \mathsf{M}(\Sigma, K)) \quad L(\mathcal{L}) = L(\mathcal{L}') \quad \Longrightarrow \quad (\forall i \in I)\ \mathsf{c}_i(\mathcal{L}) \subseteq \mathsf{c}_i(\mathcal{L}')$$

Basically, a compact IDES $\mathcal{L}$ contains the minimal set of components that generates the language $L(\mathcal{L})$ under the layout of $\mathcal{L}$. This minimal always exists as for any two IDES $\mathcal{L} = (\boldsymbol{L}, K)$ and $\mathcal{L}' = (\boldsymbol{L}', K)$ with $L(\mathcal{L}) = L(\mathcal{L}) = L$ it is easy to see that $L(\boldsymbol{L} \sqcap \boldsymbol{L}', K) = L$. We will write $\mathsf{R}_i(\boldsymbol{L}, X)$ to denoted the language vector obtained by replacing the ith component of $\boldsymbol{L}$ by the language $X \subseteq \Sigma_i^*$. Let the operator $\mathsf{C}_K : \mathsf{M}(\Sigma, K) \to \mathsf{M}(\Sigma, K)$ associate each IDES in $\mathsf{M}(\Sigma, K)$ with the matching compact IDES that generates the same language. The following proposition defines the construction of the language vector $\mathsf{C}_K(\mathcal{L})$.

Proposition 5

$$(\forall i \in I) \quad \mathsf{c}_i(\mathsf{C}_K(\mathcal{L})) = \mathsf{c}_i(\mathcal{L}) \cap P_i \|_K \mathsf{R}_i(\boldsymbol{L}, \Sigma_i^*)$$

□

Compactness is a components property which ensures that the components of a system are exactly the componentwise projections of the system language. Determinism on the other hand is a layout property which ensures that any language within the layout is decomposable (optimally) by projection. A combination of both features paves the way for the decomposition of the verification problem as expressed in the following.

Theorem 1 Let $\mathcal{L}$ be a compact IDES with a deterministic layout over a process space $\boldsymbol{\Sigma}$ and $S \subseteq \alpha(\boldsymbol{\Sigma})^*$ be a language; then

$$L(\mathcal{L}) \subseteq S \iff (\forall i \in I)\ \mathsf{c}_i(\mathcal{L}) \subseteq \mathsf{c}_i(\boldsymbol{P}(S \cap K))$$

□

Decomposition of the verification problem for compact IDES over a deterministic layout is achieved as follows. Given a compact IDES $\mathcal{L} = (\boldsymbol{L}, K)$ over $\boldsymbol{\Sigma}$ and specification language $S \subseteq \alpha(\boldsymbol{\Sigma})^*$. Clearly, $L(\mathcal{L}) \subseteq S$ if and only if $L(\mathcal{L}) \subseteq S \cap K$. If K is deterministic then the language $S \cap K$ is decomposable into the IDES $\mathcal{S} = (\boldsymbol{S}, K)$ where $\boldsymbol{S} = \boldsymbol{P}(S \cap K)$. It is easy to see that $\mathcal{S}$ is compact. As a consequence of the compactness, the inclusion of $L(\mathcal{L})$ in $L(\mathcal{S})$ is decidable by testing the inclusion of their corresponding components.

6 Supervisory Control of Interacting Discrete Event Systems

Typically, control problems require constructing a supervisor for a DES G such that the controlled behavior of G follows some predefined specifications. Based on the physical characteristics of the system, events are regarded as either controllable - can be enabled or disabled by an external agent - or uncontrollable - cannot be disabled and considered permanently enabled. For a set of events Σ, the set of controllable events is denoted Σ_c and the set of uncontrollable events is denoted Σ_u. In the linguistic setting, a language $K \subseteq \Sigma^*$ is said to be controllable with respect to L if

$$\overline{K}\Sigma_u \cap L \subseteq \overline{K}$$

The set of controllable sublanguages of L is denoted $\mathcal{C}(L)$. This set is closed under union and hence possesses a supremal element with respect to any given specification language E and this element is denoted $\sup_E \mathcal{C}(L)$. In the multi-process setting, the alphabet vector $\boldsymbol{\Sigma}$ is partitioned into a controllable vector $\boldsymbol{\Sigma}_c$ and an uncontrollable vector $\boldsymbol{\Sigma}_u$. The controllable (uncontrollable) vector contains the set of controllable (uncontrollable) events in the components of $\boldsymbol{\Sigma}$. A layout $K \in \Upsilon(\boldsymbol{\Sigma})$ is called piecewise-controllable if

$$(\forall s \in \overline{K}) \quad \mathit{Last}(s) \in \mathsf{c}_i(\boldsymbol{\Sigma}) \ \Rightarrow \ \mathit{Elig}_{\overline{K}}(s) \cap (\alpha(\boldsymbol{\Sigma}_u) - \mathsf{c}_i(\boldsymbol{\Sigma}_u))$$

The above condition states that every segment of a component i in the layout of the systems must start with a controllable event from this component, namely an event in $\mathsf{c}_i(\boldsymbol{\Sigma}_c)$. Consequently all shared events must be controllable. It is easy to check that the above condition holds if and only if the language $K \cap (\cup_{i \in I}(\Sigma - \Sigma_i)^* \Sigma_i (\Sigma_u - \Sigma_{ui}) \Sigma^*)$ is empty, where $\Sigma = \alpha(\boldsymbol{\Sigma})$ and $\Sigma_u = \alpha(\boldsymbol{\Sigma}_u)$.

Theorem 2 Let $\mathcal{L}$ and $\mathcal{L}'$ be a compact IDES with deterministic piecewise-controllable layout; then

$$L(\mathcal{L}') \in \mathcal{C}(L(\mathcal{L})) \iff (\forall i \in I)\ \mathsf{c}_i(\mathcal{L}') \in \mathcal{C}(\mathsf{c}_i(\mathcal{L}))$$

□

Hence, under a deterministic and piecewise-controllable layout, the supervisory synthesis problem for IDES is decomposable. Given an IDES $\mathcal{L} = (\boldsymbol{L}, K)$ over Σ and specification $S \subseteq \alpha(\Sigma)$, we can clearly restrict the specification to $S \cap K$ without any sacrifice in the performance. If K is deterministic then $S \cap K$ is decomposable into the IDES $\mathcal{S} = (\boldsymbol{S}, K)$ where $\boldsymbol{S} = \boldsymbol{P}(S \cap K)$. Based on the above theorem, the language $\sup_S \mathcal{CL}(\mathcal{L})$ is equal to $\|_K \sup_{S_i} \mathcal{C}(L_i)$. Where S_i and L_i are the ith components of $\boldsymbol{L}$ and $\boldsymbol{S}$ respectively.

7 Conclusion

In this paper we introduce interacting discrete event systems as a modeling and analysis tool for discrete event systems. The main feature of IDES is the explicit treatment of the system architecture. We show that under certain architectural constraints, the verification and the supervisory control problems are solvable optimally by solving the corresponding problem for the system components.

References

[1] Y. Brave and M. Heymann. Control of discrete event systems modeled as hierarchical state machines. *IEEE Trans. Autom. Control*, 38(12):1803–1819, December 1993.

[2] P. E. Caines and Y.-J. Wei. On dynamically consistent hybrid systems. *Lecture Notes in Computer Science*, 999:86–91, 1995.

[3] H. J. Genrich and K. Lautenbach. System modeling with high-level petri nets. *Theoretical Computer Science*, 13(1):109–136, January 1981.

[4] P. Gohari and W. M. Wonham. A linguistic framework for controlled hierarchical des. In *Proc. of 4th International Workshop on Discrete Event Systems*, pages 207–212, 1998.

[5] D. Harel. Statecharts: A visual formalism for complex systems. *Science of Computer Programming*, 8(3):231–274, 1987.

[6] A. Mateescu, G. Rozenberg, and A. Salomaa. Shuffle on trajectories: Syntactic constraints. Technical Report 41, Turku Centre for Computer Science, 1996.

[7] P.J. Ramadge and W.M. Wonham. Supervisory control of a class of discrete-event systems. *SIAM Journal on Control and Optimization*, 25:206–230, 1987.

[8] S. Ramaswamy and K. S. Barber. A design architecture for the modeling, analysis and design of manufacturing control software. *Int. Journal of Intelligent Control and Systems*, 2(3):485–510, 1998.

[9] B. Wang. Top-down design for RW supervisory control theory. Master's thesis, Dept. of Elec. Eng., Univ. of Toronto, Canada, 1995.

Stability Analysis for Interconnected Hybrid Systems*

Shigeru Yamamoto and Toshimitsu Ushio
Department of Systems and Human Science
Osaka University
Toyonaka, Osaka 560–8531, JAPAN
{yamamoto, ushio}@sys.es.osaka-u.ac.jp

Abstract

In this paper, we present new stability conditions for a class of large-scale hybrid dynamical systems composed of a number of interconnected hybrid subsystems. The stability conditions are given in terms of discontinuous Lyapunov functions of the stable hybrid subsystems. Furthermore, the stability conditions are represented by LMIs (Linear Matrix Inequalities) which are computationally tractable.

1 Introduction

This paper presents new stability conditions for a class of large-scale hybrid dynamical systems composed of a number of interconnected hybrid subsystems. Hybrid dynamical systems are generally those which combine continuous and discrete behavior with both continuous and discrete states variables (see, e.g., [1]–[9]). Stability for such hybrid systems have been the subject of intensive study in past few years [10]–[13]. Most of the stability results for hybrid systems extend the classical Lyapunov theory. Contrary to the conventional Lyapunov theory, we are allowed to use a discontinuous function to measure the hybrid system's energy.

The objective of this paper is to derive stability conditions for large-scale hybrid systems in terms of the discontinuous Lyapunov functions. Sufficient conditions for stability are developed in the following setting; provided the discontinuous Lyapunov functions of hybrid subsystems, we determine a discontinuous Lyapunov function of the large-scale system composed of the interconnected subsystems by using their discontinuous Lyapunov functions. The results of this paper are based on the stability theorem in the approach of [12].

For continuous interconnected dynamical systems (which are not hybrid), the stability analysis has been of interest since 70's (see, e.g., [14], [15]). Michel and Hu [16] recently generalized the results to the case of interconnected hybrid dynamical systems. By their results, stability is checked based on the so-called M-matrix condition. On the other hand, we derive stability conditions in terms of LMIs(Linear Matrix Inequalities) [17], which are computationally tractable.

This paper is organized as follows. Section 2 presents the interconnected hybrid system considered in this paper. For the interconnected hybrid system, a general stability

---

*This work was supported in part by CREST of JST(Japan Science and Technology).

condition is derived from the discontinuous Lyapunov theorem developed by Pettersson and Lennertson [11] in Section 3. The result is restated for a class of interconnected hybrid systems with linear continuous dynamics in Section 4. In addition, an LMI condition for stability is considered in Section 5. Finally, the conclusions are presented in Section 6.

2 Interconnected Hybrid System

We consider the large-scale hybrid dynamical system composed of N interconnected subsystems S_i. Each hybrid subsystem S_i is described by

$$\begin{array}{rcl} \dot{x}^i(t) & = & f_i(x^i(t), m^i(t)) \\ m^i(t) & = & \phi_i(x^i(t), m^i(t^-)) \end{array} \tag{1}$$

where $x^i \in \mathcal{R}^{n_i}$ is the continuous state, and $m^i \in \mathcal{M}_i := \{m_1^i, \ldots, m_{N_i}^i\}$ is the discrete state. The hybrid state space $\mathcal{H}_i$ is $\mathcal{R}^{n_i} \times \mathcal{M}_i$. Here, $f_i : \mathcal{H}_i \to \mathcal{R}^{n_i}$ is the continuous dynamics of (1), and $\phi_i : \mathcal{H}_i \to \mathcal{M}_i$ is the discrete dynamics of (1), respectively. Figure 1 shows the interaction between f_i and ϕ_i in a hybrid system where $N = 1$. The discrete state $m_i^i(t)$ is assumed to be piecewise continuous from the right (the notation t^- indicates $m(t^-) := \lim_{h \uparrow 0} m(t+h)$). Hence, the continuous state trajectory $x^i(\cdot)$ evolves according to $\dot{x}^i(t) = f_i(x^i(t), m^i(t))$ until $m^i(\cdot)$ changes when $x^i(\cdot)$ hits given boundaries. The boundaries can be expressed by sets

$$S_{rq}^i = \{x^i \in \mathcal{R}^{n_i} \mid m_q^i = \phi_i(x^i, m_r^i)\}. \tag{2}$$

Figure 2 illustrates a simple example of a trajectory of a hybrid system, where $N = 1$, with two discrete state $\mathcal{M} = \{m_1, m_2\}$ and the continuous state $x \in \mathcal{R}^2$.

Furthermore, assume that the equilibrium point of the continuous dynamics of (1) is $x^i = 0$. Hence, the equilibrium state of (1) is $(0, \underline{m}^i)$ where $\underline{m}^i$ satisfies $\underline{m}^i = \phi_i(0, \underline{m}^i)$.

The interconnected hybrid system under consideration is assumed to be composed of N subsystems S_i, which can be expressed by

$$\begin{array}{rcl} \dot{x}(t) & = & f(x(t), m(t)) \\ m(t) & = & \phi(x(t), m(t^-)) \end{array} \tag{3}$$

where

$$x = \begin{bmatrix} x^1 \\ \vdots \\ x^N \end{bmatrix}, \quad m = \begin{bmatrix} m^1 \\ \vdots \\ m^N \end{bmatrix} \tag{4}$$

$$f(x, m) = \begin{bmatrix} f_1(x^1, m^1) + \delta f_1(x^2, \ldots, x^N) \\ \vdots \\ f_N(x^N, m^N) + \delta f_N(x^1, \ldots, x^{N-1}) \end{bmatrix} \tag{5}$$

$$\phi(x, m) = \begin{bmatrix} \phi_1(x^1, m^1) \\ \vdots \\ \phi_N(x^N, m^N) \end{bmatrix}. \tag{6}$$

Here, $\delta f_i(x^1, \ldots, x^{i-1}, x^{i+1} \ldots x^N) : \mathcal{R}^{n_1} \times \cdots \times \mathcal{R}^{n_{i-1}} \times \mathcal{R}^{n_{i+1}} \times \cdots \times \mathcal{R}^{n_N} \to \mathcal{R}^{n_i}$ is the interaction between the ith and the other subsystems.

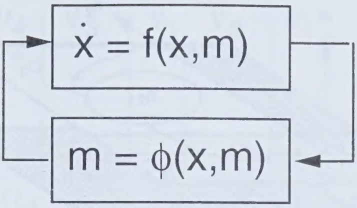


Figure 1: A hybrid dynamical system

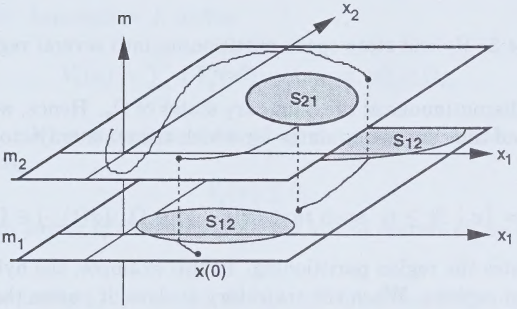


Figure 2: An example of hybrid state space and a state trajectory

3 General Stability Results

In this section, we present stability conditions for the interconnected system (3). In order to show stability, we assume that each hybrid state space $\mathcal{H}_i = \mathcal{R}^{n_i} \times \mathcal{M}_i$ is partitioned into l_i disjoint regions Ω^i_q, $q \in \mathcal{I}^i := \{1, 2, \ldots, l_i\}$. Then, let the region Ω_q of the interconnected system be defined by

$$\Omega_q := \Omega^1_{q^1} \times \Omega^2_{q^2} \times \cdots \times \Omega^N_{q^N} \tag{7}$$

$$q := \begin{bmatrix} q^1 \\ q^2 \\ \vdots \\ q^N \end{bmatrix}, \quad q^i \in \mathcal{I}^i = \{1, 2, \ldots, l_i\}. \tag{8}$$

In each region Ω_q, $q \in \mathcal{I} := \mathcal{I}^1 \times \mathcal{I}^2 \times \cdots \times \mathcal{I}^N$, a scalar continuous function $V_q(x)$ is used as a measure of the interconnected hybrid system's energy. It is assumed that each $V_q(x)$, $q \in \mathcal{I}$, is continuously differentiable on Ω_q. Hence, the time derivative of $V_q(x)$ can be written as

$$\dot{V}_q(x) = \frac{\partial V_q(x)}{\partial x} f(x, m).$$

Then, let the overall energy be given by

$$V(x) = V_q(x) \quad \text{when } (x, m) \in \Omega_q. \tag{9}$$

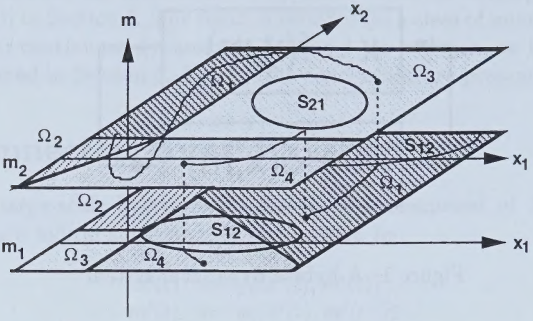


Figure 3: Hybrid state space partitioning into several regions

This $V(x)$ may be discontinuous at the boundary states of Ω_q. Hence, we define switching region Λ as the set of the continuous states for which the state trajectory passes from Ω_q to Ω_r, that is,

$$\Lambda_{qr} := \{x \mid \exists t \geq t_0 \text{ such that } (x(t^-), \cdot) \in \Omega_q, (x(t), \cdot) \in \Omega_r\}. \tag{10}$$

Figure 3 illustrates the region partitioning. In this example, the hybrid state space is partitioned into four regions. When the trajectory evolves, it passes the different regions Ω_4, Ω_3, Ω_1, Ω_3, Ω_1 and so on.

Assumption 1 *Each hybrid subsystem S_i (1) has a scalar function $V_q^i(x)$, where each $V_q^i(x)$ is differentiable in $(x, \cdot) \in \Omega_q$, satisfying the following three conditions:*

(i) $\forall q \in \mathcal{I}^i$, there exist class K functions α^i and β^i such that $\forall (x^i, \cdot) \in \Omega_q$,

$$\alpha^i(\|x^i\|) \leq V_q^i(x) \leq \beta^i(\|x^i\|);$$

(ii) $\forall q \in \mathcal{I}^i$: $\forall (x^i, m^i) \in \Omega_q^i$,

$$\dot{V}_q^i(x) \leq 0;$$

(iii) $\forall q, r \in \mathcal{I}^i$: $\forall x^i \in \Lambda_{qr}^i$,

$$V_q^i(x) \geq V_r^i(x).$$

Due to Corollary 1 in [12], the above assumption implies that each subsystem S_i is stable in the sense of Lyapunov.

Lemma 1 *Under Assumption 1, each hybrid subsystem S_i in (1) is stable in the sense of Lyapunov.*

(Proof) See [11].

□

Figure 4 depicts a sequence of Lyapunov functions satisfying the conditions in Assumption 1.

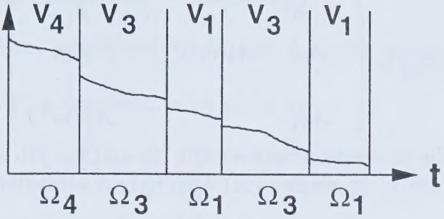


Figure 4: A sequence of Lyapunov functions

Theorem 1 *Under Assumption 1, define*

$$V_q(x) = \sum_{i=1}^{N} d^i V_q^i(x^i) \quad \textit{when } (x, m) \in \Omega_q, \tag{11}$$

where d^i $(i = 1, \ldots, N)$ are positive scalars. If there exist N scalars $d^i > 0$ such that $\forall q \in \mathcal{I}$, $\forall (x, m) \in \Omega_q$:

$$\dot{V}_q(x) \leq 0, \tag{12}$$

then the equilibrium points of the interconnected hybrid system (3) is stable in the sense of Lyapunov.

(Proof) The inequality (12) ensures the second condition of Corollary 1 in [12]. By using $d^i > 0$ satisfying (12), it follows from (i) in Assumption 1 and (11) that $\forall q \in \mathcal{I}$, $\forall (x, \cdot) \in \Omega_q$:

$$\sum_{i=1}^{N} d^i \alpha^i(\|x^i\|) \leq V_q(x) \leq \sum_{i=1}^{N} d^i \beta^i(\|x^i\|). \tag{13}$$

We define two scalar functions as

$$\alpha(\|x\|) := \sum_{i=1}^{N} d^i \alpha^i(\|x^i\|) \tag{14}$$

$$\beta(\|x\|) := \sum_{i=1}^{N} d^i \beta^i(\|x^i\|). \tag{15}$$

Then, it is obviously that $\alpha(\cdot)$ and $\beta(\cdot)$ are class K functions. Hence, the first condition of Corollary 1 in [12] is satisfied. Finally, from (iii) in Assumption 1 and (11) we can show that $\forall q, r \in \mathcal{I}$: $\forall x \in \Lambda_{qr}$,

$$V_q(x) \geq V_r(x). \tag{16}$$

This inequality is the third condition of Corollary 1 in [12]. Hence, we conclude that the interconnected hybrid system (3) is stable. □

4 Linear Case

In this section, we consider a special class of (3), that is, the continuous dynamics is linear, which is expressed by

$$\begin{aligned} \dot{x}(t) &= A(m(t))x(t) \\ m(t) &= \phi(x(t), m(t^-)) \end{aligned} \tag{17}$$

where

$$A(m) = \begin{bmatrix} A_1(m^1) & A_{12} & \cdots & A_{1N} \\ A_{21} & A_2(m^2) & \ddots & A_{2N} \\ \vdots & \ddots & \ddots & \vdots \\ A_{N1} & \cdots & \cdots & A_N(m^N) \end{bmatrix} \tag{18}$$

Here, $A_{ij} \in \mathcal{R}^{n_i \times n_j}$ is the interaction between the ith and the jth subsystems.

Similar to Assumption 1, we assume that each hybrid subsystem S_i

$$\begin{array}{rcl} \dot{x}^i(t) & = & A_i(m^i(t))x^i(t) \\ m^i(t) & = & \phi_i(x^i(t), m^i(t^-)) \end{array} \tag{19}$$

is stable. To show the stability, hereafter we define the energy $V_q^i(x^i)$ of the hybrid subsystem as a quadratic function $x^{i^T} P_q^i x^i$ by using a symmetric matrix P_q^i. When we define $V_q(x^i)$ as $x^{i^T} P_q^i x^i$, Assumption 1 can be recast into the following assumption.

Assumption 2 *Each hybrid subsystem S_i (19) has a symmetric matrix P_q^i satisfying the following three conditions:*

(i) $\forall q \in \mathcal{I}^i$, $\exists \alpha^i > 0$, $\exists \beta^i > 0 : \forall (x^i, \cdot) \in \Omega_q$,

$$\alpha^i \|x^i\|^2 \le x^{i^T} P_q^i x^i \le \beta^i \|x^i\|^2;$$

(ii) $\forall q \in \mathcal{I}^i$: $\forall (x^i, m^i) \in \Omega_q^i$,

$$x^{i^T} P_q^i A_i(m^i) x^i \le 0;$$

(iii) $\forall q, r \in \mathcal{I}^i$: $\forall x^i \in \Lambda_{qr}^i$,

$$x^{i^T} P_q^i x^i \ge x^{i^T} P_r^i x^i.$$

Lemma 2 *Under Assumption 2, each hybrid subsystem S_i is stable with the Lyapunov function $V_q^i(x^i) = x^{i^T} P_q^i x^i$. Moreover, this $V_q^i(x^i)$ satisfies three conditions of Assumption 1.*

(Proof) See [11].

□

Corollary 1 *Under Assumption 2, if there exist N scalars $d^i > 0$ ($i = 1, \ldots, N$) such that $\forall q \in \mathcal{I}$, $\forall (x, m) \in \Omega_q$:*

$$x^T P_q A(m) x \le 0, \tag{20}$$

then the equilibrium points of the interconnected hybrid system (17) is stable in the sense of Lyapunov, where

$$P_q = \mathrm{diag}\{d^i P_q^i \mid i = 1, \ldots N\}.$$

(Proof) If we define $V_q^i(x^i)$ as $x^{i^T} P_q^i x^i$, $\forall (x, \cdot) \in \Omega_q$,

$$V_q(x) := x^T P_q x = \sum_{i=1}^{N} d^i x^{i^T} P_q^i x^i = \sum_{i=1}^{N} d^i V_q(x^i)$$

Then, from (20),

$$\dot{V}_q(x) = 2x^T P_q A(m) x \le 0$$

Therefore, we can choose $V_q(x)$ in Theorem 1 as $x^T P_q x$.

□

5 LMI Condition for Stability

To derive LMI conditions, we express V_q, Ω_q and Λ_{qr} with quadratic functions for subsystems S^i as follows.

Lyapunov function V_q is supposed to be given by

$$\begin{aligned} V_q^i &= x^{i^T} P_q^i x^i + 2{p_q^i}^T x^i + \pi_q^i \\ &= \hat{x}^{iT} \hat{P}_q^i \hat{x}^i \end{aligned} \tag{21}$$

where

$$\hat{x}^i = \begin{bmatrix} x^i \\ 1 \end{bmatrix} \in \mathcal{R}^{n_i+1}, \tag{22}$$

$$P_q^i = P_q^{iT} \in \mathcal{R}^{n_i \times n_i} \tag{23}$$

$$\hat{P}_q^i = \begin{bmatrix} P_q^i & p_q^i \\ {p_q^i}^T & \pi_q^i \end{bmatrix} \in \mathcal{R}^{(n_i+1)\times(n_i+1)} \tag{24}$$

The domain Ω_q on which V_q is valid is defined by

$$\Omega_q^i = \{x^i \in \mathcal{R}^{n_i} \mid \hat{x}^{iT} \hat{Q}_q^i(k) \hat{x}^i \geq 0,\ k = 1, \ldots, s_q^i\} \tag{25}$$

where $Q_q^i(k) = Q_q^i(k)^T \in \mathcal{R}^{n_i \times n_i}$ and

$$\hat{Q}_q^i(k) = \begin{bmatrix} Q_q^i(k) & q_q^i(k) \\ q_q^i(k)^T & \rho_q^i(k) \end{bmatrix} \in \mathcal{R}^{(n_i+1)\times(n_i+1)} \tag{26}$$

The switching region Λ_{qr} which is associated with Ω_q and Ω_r is specified by

$$\Lambda_{qr}^i = \{x^i \in \mathcal{R}^{n_i} \mid \hat{x}^{iT} \hat{R}_{qr}^i(k) \hat{x}^i \geq 0,\ k = 1, \ldots, s_{qr}^i\} \tag{27}$$

where $R_{qr}^i(k) = R_{qr}^i(k)^T \in \mathcal{R}^{n_i \times n_i}$ and

$$\hat{R}_{qr}^i(k) = \begin{bmatrix} R_{qr}^i(k) & r_{qr}^i(k) \\ r_{qr}^i(k)^T & \rho_{qr}^i(k) \end{bmatrix} \in \mathcal{R}^{(n_i+1)\times(n_i+1)} \tag{28}$$

Assumption 3 *Each hybrid subsystem S_i (19) has a symmetric matrix $\hat{P}_q^i$, $q \in \mathcal{I}^i := \{1, 2, \ldots, l_i\}$, $\omega_q(k) \geq 0$ and $\lambda_{qr}(k) \geq 0$ satisfying the following three conditions:*

(i) $\forall q \in \mathcal{I}^i = \{1, 2, \ldots, l_i\}$, $\exists \alpha^i > 0$, $\exists \beta^i > 0$:

$$\begin{bmatrix} \alpha^i I & 0 \\ 0 & 0 \end{bmatrix} + \sum_{k=1}^{s_q^i} \omega_q(k) \hat{Q}_q^i(k) \leq \hat{P}_q^i \leq \begin{bmatrix} \beta^i I & 0 \\ 0 & 0 \end{bmatrix} - \sum_{k=1}^{s_q^i} \omega_q(k) \hat{Q}_q^i(k);$$

(ii) $\forall q \in \mathcal{I}^i = \{1, 2, \ldots, l_i\}$: $\forall (\cdot, m) \in \Omega_q^i$

$$\hat{A}^i(m)^T \hat{P}_q^i + \hat{P}_q^i \hat{A}^i(m) + \sum_{k=1}^{s_q^i} \omega_q(k) \hat{Q}_q^i(k) \leq 0,$$

where

$$\hat{A}^i(m) = \begin{bmatrix} A^i(m) & 0 \\ 0 & 0 \end{bmatrix}; \tag{29}$$

(iii) $\forall q, r \in \mathcal{I}^i = \{1, 2, \ldots, l_i\}$:

$$\hat{P}_q^i - \sum_{k=1}^{r_{qr}^i} \lambda_{qr}(k) \hat{R}_{qr}^i(k) \geq \hat{P}_r^i.$$

Lemma 3 *Under Assumption 3, each S_i is stable with the Lyapunov function*

$$V^i(\hat{x}^i) = \hat{x}^{iT} \hat{P}_q^i \hat{x}^i. \tag{30}$$

(Proof) See [11].

□

Theorem 2 *Under Assumption 3, define*

$$\hat{P}_q = \begin{bmatrix} d^1 P_q^1 & 0 & 0 & d^1 p_q^1 \\ 0 & \ddots & 0 & \vdots \\ 0 & 0 & d^N P_q^N & d^N p_q^N \\ d^1 p_q^{1T} & \ldots & d^N p_q^{NT} & \sum d^i \pi_q^i \end{bmatrix}, \tag{31}$$

where d^i $(i = 1, \ldots, N)$ are positive scalars. If there exist N scalars $d^i > 0$ such that $\forall q \in \mathcal{I}$, $\forall (\cdot, m) \in \Omega_q$:

$$\hat{A}(m)^T \hat{P}_q + \hat{P}_q \hat{A}(m) + \hat{Q}_q \leq 0, \tag{32}$$

where

$$\hat{A}(m) = \begin{bmatrix} A(m) & 0 \\ 0 & 0 \end{bmatrix} \tag{33}$$

and

$$\hat{Q}_q = \begin{bmatrix} \sum_{k=1}^{s_q^1} \omega_q^1(k) Q_q^1(k) & 0 & 0 & \sum_{k=1}^{s_q^1} \omega_q^1(k) q_q^1(k) \\ 0 & \ddots & 0 & \vdots \\ 0 & 0 & \sum_{k=1}^{s_q^N} \omega_q^N(k) Q_q^N(k) & \sum_{k=1}^{s_q^N} \omega_q^N(k) q_q^N(k) \\ \sum_{k=1}^{s_q^1} \omega_q^1(k) q_q^1(k)^T & \ldots & \sum_{k=1}^{s_q^N} \omega_q^N(k) q_q^N(k)^T & \sum_{i=1}^{N} \sum_{k=1}^{s_q^i} \omega_q^i(k) \rho_q^i(k) \end{bmatrix}, \tag{34}$$

then the equilibrium points of the interconnected hybrid system (17) is stable in the sense of Lyapunov.

(Proof)Define $V_q := \hat{x}^T \hat{P}_q \hat{x}$ with

$$\hat{x} = \begin{bmatrix} x \\ 1 \end{bmatrix}. \tag{35}$$

Then,

$$\dot{V}_q = \hat{x}^T \left\{ \hat{A}(m)^T \hat{P}_q + \hat{P}_q \hat{A}(m) \right\} \hat{x}. \tag{36}$$

By using the S-procedure [17] and (31), we have

$$\dot{V}_q \leq 0. \tag{37}$$

On the other hand,

$$\begin{aligned} V_q &= \sum_{i=1}^{N} \left(d^i x^{iT} P_q^i x^i + 2d^i p_q^{iT} x^i + d^i \pi_q^i \right) \\ &= \sum_{i=1}^{N} d^i V_q^i(x^i). \end{aligned} \tag{38}$$

Each $V_q^i(x^i)$ satisfies the first and third condition of Assumption 1. Hence, the equilibrium points of the interconnected hybrid system (17) is stable. □

6 Conclusions

In this paper, we have investigated stability for a class of interconnected hybrid dynamical systems. The sufficient conditions for stability are derived in terms of Lyapunov functions of hybrid subsystems. The Lyapunov functions are allowed to be discontinuous. The results are also modified to an LMI form which is computationally tractable.

The interaction δf in (3) between hybrid subsystems is dependent on only the continuous state x. This assumption may be slightly restrictive in real systems. Allowing δf to depend on the discrete state m is also a future work.

References

[1] R. L Grossman, A. Nerode, A. P. Ravn, and H. Rischel, Eds., *Hybrid Systems*, Lecture Notes in Computer Science, no. 736, New York: Springer-Verlag, 1993.

[2] P. Antsaklis, W. Kohn, A. Nerode, and S. Sastry, Eds., *Hybrid Systems II*, Lecture Notes in Computer Science, no. 999, New York: Springer-Verlag, 1994.

[3] R. Alur, T. Henzinger, and E. Sontag, Eds., *Hybrid Systems III — Verification and Control*, Lecture Notes in Computer Science, no. 1066, New York: Springer-Verlag, 1996.

[4] P. Antsaklis, W. Kohn, A. Nerode, and S. Sastry, Eds., *Hybrid Systems IV*, Lecture Notes in Computer Science, no. 1273, New York: Springer-Verlag, 1997.

[5] P. Antsaklis, W. Kohn, M. Lemmon, A. Nerode, and S. Sastry, Eds., *Hybrid Systems V*, Lecture Notes in Computer Science, no. 1567, New York: Springer-Verlag, 1998.

[6] T. A. Henzinger and S. Sastry, Eds., *Hybrid Systems: Computation and Control*, Lecture Notes in Computer Science, no. 1386, New York: Springer-Verlag, 1998.

[7] R. W. Brockett, "Hybrid models for motion control systems," *Essays in Control*, H. L. Trentleman and J. C. Willems Eds., Birkhäuser, pp. 29-53, 1993.

[8] M. S. Branicky, "A Unified Framework for Hybrid Control: Model and Optimal Control Theory," *IEEE Trans. on Automatic Control*, Vol. 43, No. 1, pp. 31–45, 1998.

[9] H. Ye, A. N. Michel and L. Hou, "Stability Theory for Hybrid Dynamical Systems," *IEEE Trans. on Automatic Control*, Vol. 43, No. 4, pp. 461–474, 1998.

[10] M. S. Branicky, "Multiple Lyapunov Functions and Other Analysis Tools for Switched and Hybrid Systems," *IEEE Trans. on Automatic Control*, Vol. 43, No. 4, pp. 475–490, 1998.

[11] S. Pettersson and B. Lennartson, "LMI for Stability and Robustness of Hybrid Systems," *Technical Report CTH/RT/I-96/005*, Control Engineering Lab., Charmers University of Technology, 1998.

[12] S. Pettersson and B. Lennartson, "LMI for Stability and Robustness of Hybrid Systems," *Proc. ACC'97*, pp. 1714–1718, 1997.

[13] M. Wicks, P. Peleties and R. DeCarlo, "Switched Controller Synthesis for the Quadratic Stabilization of a Pair of Unstable Linear Systems," *European Journal of Control*, Vol. 4, pp. 140–147, 1998.

[14] A. N. Michel and R. K. Miller, *Qualitative Analysis of Large Scale Dynamical Systems*, Mathematics in Science and Engineering, vol. 134, Academic Press, 1977.

[15] D. D. Šiljak, *Large-Scale Dynamical Systems – Stability and Structure*, North-Holland, 1978.

[16] A. N. Michel and B. Hu, "Stability Analysis of Interconnected Hybrid Dynamical Systems," *Proc. IFAC LSS '98 Large Scale Systems: Theory and Applications*, pp. 698–703, 1998.

[17] S. P. Boyd, L. E. Ghaoui, E. Feron and V. Balakrishnan, "Linear Matrix Inequalities in System and Control Theory," SIAM, Philadelphia, 1994.

Decentralized Supervisory Control of Concurrent Discrete Event Systems with Partial Observations *

Shengbing Jiang and Ratnesh Kumar
Department of Electrical Engineering
University of Kentucky
Lexington, KY 40506-0046
{sjian0, kumar}@engr.uky.edu

Abstract

The decentralized supervisory control problem of concurrent discrete event systems with partial observations is studied in this paper. A necessary and sufficient condition is obtained for the existence of decentralized supervisors for ensuring that the controlled behavior of the system lies in a given range. For illustration an example of a simple manufacturing system is given. This paper extends the results of [1, 9, 6, 7, 10] on concurrent systems and partial observations.

1 Introduction

The supervisory control theory for discrete event systems was first proposed by Ramadge and Wonham. In [8], they introduced the notion of controllability as a necessary and sufficient condition for the existence of a supervisor that achieves a desired controlled behavior for a given discrete event system under the complete observation of events. When the events are not completely observed by the supervisor and are filtered by an observation mask, an addition condition of observability introduced by Lin and Wonham [5], and Cieslak *et al.* [1] is needed for the existence of the supervisor. In the more general case of decentralized control, when there are more than one supervisors that the uncontrollable event set for the ith supervisor is Σ_{ui}, and its observation is filtered through the mask M_i, the condition of co-observability is required in place of observability, as shown Cieslak *et al.* [1] for the case when the controlled behavior is given as a prefix-closed language, and later generalized to the non-prefix-closed case by Rudie and Wonham [9].

In [6], Lin and Wonham considered a specialized version of the decentralized control, where the mask M_i is a projection type mask, and the local controllable event set Σ_{ci} is a subset of local observable event set $\Sigma_i := M_i(\Sigma)$. Under such assumptions, the authors gave a sufficient condition in terms of the normality that guarantees that the decentralized control can achieve the optimal behavior achievable by a centralized supervisor. Later in [7], they extended their results to the systems with partial observation, where the local event set Σ_i is the union of the local controllable event set and the local observable event set, *i.e.*, $\Sigma_i = \Sigma_{ci} \bigcup \Sigma_{oi}$, and the local supervisor S_i is constructed by using only the local information $T_i(L(G))$, where T_i is the projection mask from the global event set Σ to the local event set Σ_i, and G is the state machine representing the discrete event

---

*This research was supported by the National Science Foundation under the Grant NSF-ECS-9709796.

system under control. They gave a sufficient condition for the existence of decentralized supervisors that ensure that the controlled behavior of the system lies in a given range expressed by local specifications.

In [10], Willner and Heymann also studied the problem of [6], except that the systems they considered were concurrent systems. They introduced the notion of separability, and under the assumption that $\Sigma_{ui} \bigcap \Sigma_j = \emptyset$ for all $i \neq j$ they proved that the separability is a necessary and sufficient condition that guarantees that the decentralized control can achieve the optimal behavior achievable by a centralized supervisor.

In this paper, we first examine the problem as in [7], shown in Figure 1, except we relax the assumption that the local event set Σ_i is the union of the local controllable event set and the local observable event set. We obtain a necessary and sufficient condition for the existence of decentralized supervisors that ensure that the controlled behavior of the system lies in a given range, extending the result of [7] where only a sufficient condition was given for a more constrained problem.

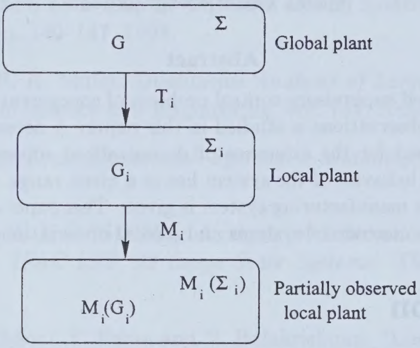


Figure 1: Diagram of the system

Next we consider the decentralized control of concurrent systems with partial observations, which is a special case of the above problem with $L(G) = \cap_i T_i^{-1}[L(G_i)]$, $G = \|_i G_i$. A necessary and sufficient condition is also derived for the existence of decentralized supervisors. Finally, an illustrative example based on a simple manufacturing system is provided.

This paper extends the results of [1, 9, 6, 7, 10] on concurrent systems and partial observations in the following ways:

1. The problems studied in [1, 9] are special cases of the problems considered in this paper, and their results can be derived from our results as corollaries. For the concurrent systems, our results also result in computational savings in the synthesis of the decentralized supervisors compared with the results of [1, 9].

2. In this paper, partial observation masks are introduced for local systems, which were not considered in [6, 10]. Even for the similar problem studied in [6, 10], we provide a necessary and sufficient condition for the existence of decentralized supervisors, where [6] only gave a sufficient condition, and [10] gave the same necessary and sufficient condition as ours under a more restricted assumption.

3. For the same problem studied in [7], we give a necessary and sufficient condition for the existence of decentralized supervisors, where [7] only gave a sufficient condition for a more restricted system.

2 Background Review

This paper is set in the supervisory control framework for discrete-event systems developed by Ramage and Wonham [8]. For the readers' convenience, some background results from the cited references are first provided in this section. For a detailed introduction of the theory, readers may refer to the book [2].

An uncontrolled discrete event system is modeled as an automaton

$$G = (Q, \Sigma, \delta, q_0, Q_m)$$

where Σ is a set of event labels, Q is a set of states, $q_0 \in Q$ is the initial state, $Q_m \subseteq Q$ is the set of marked states, and $\delta : \Sigma \times Q \to Q$, the transition function, is a partial function defined at each state in Q for a subset of Σ. Let Σ^* denote the set of all finite strings over Σ including the null string ϵ. Then $\delta : \Sigma \times Q \to Q$ can be extended in the obvious way to $\delta : \Sigma^* \times Q \to Q$. Then the language generated by G is $L(G) := \{s \in \Sigma^* \mid \delta(s, q_0) \text{ is defined}\}$, and the language marked by G is $L_m(G) := \{s \in \Sigma^* \mid \delta(s, q_0) \in Q_m\}$.

In general, a language over Σ is any subsut of Σ^*. The prefix-closure $pr(K)$ of a language $K \subseteq \Sigma^*$ is the set of all prefixes of strings in K. K is called prefix-closed if $pr(K) = K$.

To impose supervision on the system, the event set Σ is partitioned into two subsets Σ_c and Σ_u of controllable and uncontrollable events respectively. A supervisor S is a pair (R, ψ) where R is an automaton which recognizes a language over the same event set as G, and ψ is the feedback map from the event set and the states of R to the set $\{\textit{enable, disable}\}$. If X denotes the set of states of R, then $\psi : \Sigma \times X \to \{\textit{enable, disable}\}$ satisfies: $\psi(\sigma, x) = \textit{enable}$ if $\sigma \in \Sigma_u$. R is considered to be driven by the strings generated by G, and in turn, at each state of R, the control rule $\psi(\sigma, x)$ dictates the occurence of σ at the corresponding state of G. The behavior of the supervised system is represented by an automaton S/G. The language generated by the supervised system is denoted by $L(S/G)$, and its marked language is $L_m(S/G) := L(S/G) \cap L_m(G)$.

A supervisor $S = ((X, \Sigma, \xi, x_0), \psi)$ is complete for G if for every string s in Σ^* and every event σ in Σ, the conditions $s \in L(S/G), s\sigma \in L(G)$, and $\psi(\sigma, \xi(s, x_0)) = \textit{enable}$ together imply that $s\sigma \in L(S/G)$. It will always be assumed that S is complete.

Given a nonempty prefix-closed sublanguage K of $L(G)$, there exists a supervisor S such that $L(S/G) = K$ iff K is $(L(G), \Sigma_u)-$controllable, *i.e.*, $pr(K)\Sigma_u \bigcap L(G) \subseteq pr(K)$. If K is not controllable, then a supervisor is synthesized for achieving the supremal controllable sublanguage of K, denoted by $sup\overline{C}(K)$.

When the supervisor can not observe all the events, the concept of observation mask is introduced. An observation mask is a function $M : \Sigma \to \Delta \bigcup \{\epsilon\}$, where $\epsilon \notin \Delta$, and Δ is called the set of observed events. The mask function can be extended to the set of strings in a natural way. A supervisor S for G is said to be mask-compatible if it observes only $M(L(G))$. In this paper, it is assumed that all supervisors for partially observed systems must be mask-compatible supervisors.

A prefix-closed sublanguage K of $L(G)$ is said to be $(L(G), M)-$observable if the conditions $(\forall s, s' \in K) M(s) = M(s'), (\forall \sigma \in \Sigma) s\sigma \in K$, and $s'\sigma \in L(G)$ together imply that $s'\sigma \in K$. If K is not observable, then the infimal observable superlanguage of K exists, which is denoted by $inf\overline{O}(K)$.

Given a nonempty prefix-closed sublanguage K of $L(G)$ and an observation mask M, there exists a supervisor S such that $L(S/G) = K$ if and only if K is $(L(G), \Sigma_u)$-controllable and $(L(G), M)$-observable.

Let $A, E \subseteq \Sigma^*$ be two languages with $A \subseteq E \subseteq L_m(G)$, M be an observation mask, then there exists a supervisor S such that $A \subseteq L(S/G) \subseteq E$ if and oly if $inf\overline{PCO}(A) \subseteq E$, where $inf\overline{PCO}(A)$ is the infimal prefix-closed, controllable, and obsevable superlanguage of A.

3 Decentralized Control

In this section we study the decentralized supervisory control problem of concurrent discrete event systems with partial observations. We first consider the problem studied in [7], but without the assumption that the local event set is the union of the local controllable event set and the local observable event set, and the observation mask is not necessarily the projection type.

Let the global event set Σ be the union of local event sets $\Sigma = \cup_i \Sigma_i$, $i \in I$. Let $T_i : \Sigma \to \Sigma_i \cup \{\epsilon\}$ denote the natural projection from the global event set to the local event set. Let Σ_c be the global cotrollable event set, then the ith local controllable event set is given by $\Sigma_{ci} := \Sigma_c \cap \Sigma_i$. Finally, let $M_i : \Sigma_i \to \Delta_i \bigcup \{\epsilon\}$ be the local observation mask. We use G_i to denote the automaton with $L(G_i) = T_i[L(G)]$, $i \in I$; here $L(G_i)$ represents the ith local behavior of the system G. If S_i is a local supervisor for G_i, then we can extend S_i to a global supervisor $\tilde{S}_i$ for G, by letting $\tilde{S}_i$ not observe all events in $\Sigma - \Sigma_i$, and permanently enable all events in $\Sigma_c - \Sigma_{ci}$. Then it is easy to see that $L(\tilde{S}_i/G) = L(G) \bigcap T_i^{-1}[L(S_i/G_i)]$, $i \in I$.

Next we give the definition of *co-normal*, which is similar as the *decomposable* in [9]. We prefer the name of *co-normal*, since it reflects its relation with *normal* just like the relation between *observable* and *co-observable.* For a language $K \subseteq L(G)$, K is $(G, \{T_i, i \in I\})$ co-normal if there exists a group of languages $\{K_i \subseteq T_i[L(G)], i \in I\}$ such that $K = L(G) \bigcap (\cap_i T_i^{-1}(K_i))$. Note that this is a natural generalization of the notion of normality to the decentralized setting. Further when $L(G) = \Sigma^*$, so that $K = \cap_i T_i^{-1}(K_i)$, K is said to be separable.

Now we consider the "target specification" problem:

> Given a nonempty prefix-closed sublanguage K of $L(G)$, find the local supervisors $\{S_i,\ i \in I\}$ such that $L((\bigwedge_i \tilde{S}_i)/G) = K$, *i.e.*, $L(G) \bigcap (\cap_i T_i^{-1}[L(S_i/G_i)])$ $= K$.

For the existence of the local supervisors, a necessary and sufficient condition is obtained in the following theorem. Before giving the theorem, we first give a lemma which is needed in the proof of the theorem. This lemma is similar to [10, Corollary 2.1] which applied to the separability.

Lemma 1 Let G be a plant, Σ be the global event set, which is the union of local event sets, *i.e.*, $\Sigma = \cup_i \Sigma_i$, $i \in I$. Let T_i be the natural projection from Σ to Σ_i. A language $K \subseteq L(G)$ is $(G, \{T_i, i \in I\})$ co-normal if and only if $K = L(G) \bigcap (\cap_i T_i^{-1}[T_i(K)])$.

Now we present the theorem:

Theorem 1 Let G be a plant, Σ be the global event set, which is the union of local event sets, *i.e.*, $\Sigma = \cup_i \Sigma_i$, $i \in I$. Let T_i be the natural projection from Σ to Σ_i, Σ_c be the global controllable event set, $\Sigma_{ci} := \Sigma_c \cap \Sigma_i$ be the local controllable event set, M_i be the local observation mask. Given a nonempty prefix-closed sublanguage K of $L(G)$, there exists a group of local supervisors $\{S_i, i \in I\}$ such that $L((\bigwedge_i \tilde{S}_i)/G) = K$ if and only if

$$K = L(G) \bigcap (\cap_i T_i^{-1}[inf\overline{CO}_i(T_i(K))]),$$

where $\tilde{S}_i$ is the extension of S_i to the global system G, and $inf\overline{CO}_i(T_i(K))$ is the infimal $(L(G_i), \Sigma_{ui})-$controllable and $(L(G_i), M_i)-$observable superlanguage of $T_i(K)$, and $L(G_i) := T_i[L(G)]$ is the ith local behavior of the system, $\Sigma_{ui} := \Sigma_i - \Sigma_{ci}$ is the local uncontrollable event set.

Remark 1 If the projections $\{T_i,\ i \in I\}$ in Theorem 1 are identity projections, *i.e.*, $\forall i \in I$, $\Sigma_i = \Sigma$, then the necessary and sufficient condition in Theorem 1 becomes $K = \cap_i[inf\overline{CO}_i(K)]$. From [3, Theorem 3], we know this is equivalent to K being $(L(G), \Sigma_u)-$controllable ($\Sigma_u = \cup_i \Sigma_{ui}$), and $(L(G), \{\Sigma_{ui}\}, \{M_i\})$-co-observable, which is the result obtained in [1, 9]. So Theorem 1 extends the result of [1, 9]. A modular computation for $inf\overline{CO}_i(K)$ was obtained in [3].

Remark 2 If we introduce a new observation mask $\tilde{M}_i : \Sigma \to \Delta \cup \{\epsilon\}$ such that $\tilde{M}_i = M_i T_i$, we can use the result in [1, 9] to directly find the decentralized supervisors $\{\tilde{S}_i\}$ for G, rather than the approach of Theorem 1, which first finds the local supervisors $\{S_i\}$ for the local systems $\{G_i\}$. To obtain $\tilde{S}_i$ for G directly, we need $L(\tilde{S}_i/G) = inf\overline{CO}_i(K)$, where $inf\overline{CO}_i(K)$ is the infimal $(L(G), \Sigma_{ui})$-controllable and $(L(G), \tilde{M}_i)$-observable superlanguage of K. This, however, uses the global information $L(G)$. On the other hand, the approach of Theorem 1 computes the local supervisors $\{S_i\}$ for local systems $\{G_i\}$, such that $L(S_i/G_i) = inf\overline{CO}_i(T_i(K))$, which only requires the local information $L(G_i) = T_i[L(G)]$. So when G is given as a concurrent system, $G = \|_i G_i$, result of Theorem 1 would result in computational savings in the synthesis of the decentralized supervisors.

Next we consider a special case of the problem in Theorem 1 as considered in [6]. Suppose the observation mask M_i for each local supervisor is identity (this is the setting of [6]). Then from Theorem 1 we have the following corollary:

Corollary 1 Under the same assumptions given in Theorem 1, except that each local observation mask M_i is identity, then for a given nonempty prefix-closed sublanguage K of $L(G)$, there exist local supervisors $\{S_i,\ i \in I\}$ such that $L((\bigwedge_i \tilde{S}_i)/G) = K$ if and only if $K = L(G) \bigcap (\cap_i T_i^{-1}[inf\overline{C}_i(T_i(K))])$.

We show in the next lemma that the condition in Corollary 1 is equivalent to the condition that K is globally controllable and co-normal, thereby strengthening the result of the corollary.

Lemma 2 $K = L(G) \bigcap (\cap_i T_i^{-1}[inf\overline{C}_i(T_i(K))])$ if and only if K is $(L(G), \Sigma_u)$-controllable and $(G, \{T_i, i \in I\})$ co-normal, where $\Sigma_u := \Sigma - \Sigma_c$ is the global uncontrollable event set.

From Corollary 1 and Lemma 2, we have the following theorem:

Theorem 2 Let G be a plant, Σ be the global event set, which is the union of local event sets, *i.e.*, $\Sigma = \cup_i \Sigma_i$, $i \in I$. Let T_i be the natural projection from Σ to Σ_i, Σ_c be the global controllable event set, $\Sigma_{ci} := \Sigma_c \cap \Sigma_i$ be the local controllable event set, M_i be the local observation mask. If each observation mask M_i is identity then for a given nonempty prefix-closed sublanguage K of $L(G)$, there exist local supervisors $\{S_i,\ i \in I\}$ such that $L((\bigwedge_i \tilde{S}_i)/G) = K$ if and only if K is $(L(G), \Sigma_u)$-controllable and $(G, \{T_i, i \in I\})$ co-normal, where $\tilde{S}_i$ is the extension of S_i to the global system G.

Remark 3 In [6], the following problem was considered:

> Given a prefix-closed language $E = \cap_i T_i^{-1}(E_i)$, let $K = sup\overline{C}(L(G) \bigcap E)$. Determine a condition that ensures that $K = L((\bigwedge_i \tilde{S}_i)/G)$ for the local supervisors $\{S_i,\ i \in I\}$ with $L(S_i/G_i) = sup\overline{C}_i(L(G_i) \cap E_i)$.

The authors gave a sufficient condition that $sup\overline{C}(L(G) \cap T_i^{-1}(E_i))$ is $(L(G), T_i)$-normal for each $i \in I$ [6, Theorem 4.1]. In fact, the above problem consists of two sub-problems: First one is to find the local supvisors $\{S_i\}$ for local systems $\{G_i\}$ such that $K = L((\bigwedge_i \tilde{S}_i)/G)$, and the second one is to determine a condition under which the ith local supervisor S_i can be designed from the ith local specification E_i, *i.e.*, $L(S_i/G_i) = sup\overline{C}_i(L(G_i) \cap E_i)$. If E is given as a global specification, then the first sub-problem is the only one we need to solve. Theorem 2 gives a necessary and sufficient condition for this problem.

Having solved the "target specification" problem, we next consider the following "range specification" problem:

> Given two languages A and E with $A \subseteq E \subseteq L(G)$, find local supvisors $\{S_i, i \in I\}$ such that $A \subseteq L((\bigwedge_i \tilde{S}_i)/G) \subseteq E$.

This problem should be viewed as a generalization of the "target specification" problem studied above, since when $A = E = K$, it reduces to the target specification problem. The following theorem provides a necessary and sufficient condition for the "range specification" problem.

Theorem 3 Let G be a plant, Σ be the global event set, which is the union of local event sets, *i.e.*, $\Sigma = \cup_i \Sigma_i$, $i \in I$. Let T_i be the natural projection from Σ to Σ_i, Σ_c be the global controllable event set, $\Sigma_{ci} := \Sigma_c \cap \Sigma_i$ be the local controllable event set, M_i be the local observation mask. Given two languages A and E with $A \subseteq E \subseteq L(G)$, there exist local supervisors $\{S_i, i \in I\}$ such that $A \subseteq L((\bigwedge_i \tilde{S}_i)/G) \subseteq E$ if and only if

$$L(G) \bigcap (\cap_i T_i^{-1}[inf\overline{PCO}_i(T_i(A))]) \subseteq E,$$

where $inf\overline{PCO}_i(T_i(A))$ is the infimal prefix-closed, $(L(G_i), \Sigma_{ui})$−controllable, and $(L(G_i), M_i)$−observable superlanguage of $T_i(A)$.

Remark 4 As in Remark 1, if the projections $\{T_i,\ i \in I\}$ in Theorem 3 are identity projections, *i.e.*, $\forall i \in I$, $\Sigma_i = \Sigma$, then the necessary and sufficient condition in Theorem 3 becomes $\cap_i[inf\overline{PCO}_i(A)] \subseteq E$. From [3], we know this is equivalent to being $inf\overline{PCCo}(A) \subseteq E$, where $inf\overline{PCCo}(A)$ is the infimal prefix-closed, $(L(G), \Sigma_u)$-controllable, and $(L(G), \{\Sigma_{ui}\}, \{M_i\})$ co-observable superlanguage of A, which is the result obtained in [9, Theorem 4.2]. So Theorem 3 extends the result of [9].

If A and E are given by the local specifications, then we can derive the following corollary from Theorem 3:

Corollary 2 Let $A = L(G)\bigcap(\cap_i T_i^{-1}A_i)$ and $E = L(G)\bigcap(\cap_i T_i^{-1}E_i)$ with $A_i \subseteq E_i \subseteq L(G_i)$. If $inf\overline{PCO}_i(A_i) \subseteq E_i$ for each i, then there exist local supervisors $\{S_i, i \in I\}$ such that $A \subseteq L((\bigwedge_i \tilde{S}_i)/G) \subseteq E$.

Remark 5 In [7, Theorem 5], the authors studied the same problem for which Corollary 2 provides a solution. The sufficient condition given there implies the condition given in Corollary 2. So we are able to extend the result of [7].

So far, we have not assumed any structure properties of the system considered. Now we suppose that the system G is composed of concurrent subsystems, G_i, with event sets Σ_i, $i \in I$, *i.e.*, $G :=\|_i G_i$. From [10, Proposition 3.1], we have $L(G) = \cap_i T_i^{-1}[L(G_i)]$. (Note that in an earlier analysis we also had the local systems $\{G_i\}$ with $L(G_i) = T_i L(G)$, but we did not have the property that $L(G) = \cap_i T_i^{-1}[L(G_i)]$, which holds for the concurrent system.) We also note that for the concurrent system the property $L(G_i) \supseteq T_i L(G)$ holds instead of the definition $L(G_i) := T_i L(G)$ used for non-concurrent systems. It can be seen that a concurrent system is a special case of the systems we considered up to now. So we can derive the results for the concurrent system as before, except applying that $L(G) = \cap_i T_i^{-1}[L(G_i)]$. We first give a result similar to that of Theorem 1.

Theorem 4 Let G be a concurrent system with $G =\|_i G_i$, $i \in I$, Σ be the global event set, which is the union of local event sets, *i.e.*, $\Sigma = \cup_i \Sigma_i$, $i \in I$. Let T_i be the natural projection from Σ to Σ_i, Σ_{ci} be the local controllable event set, M_i be the local observation mask. Then for a given nonempty prefix-closed sublanguage K of $L(G)$, there exist local supervisors $\{S_i, i \in I\}$ for $\{G_i, i \in I\}$ such that $\cap_i T_i^{-1}[L(S_i/G_i)] = K$ if and only if $K = \cap_i T_i^{-1}[inf\overline{CO}_i(T_i(K))]$, where $inf\overline{CO}_i(T_i(K))$ is the infimal $(L(G_i), \Sigma_{ui})$-controllable and $(L(G_i), M_i)$-observable superlanguage of $T_i(K)$, and $\Sigma_{ui} := \Sigma_i - \Sigma_{ci}$ is the local uncontrollable event set.

Remark 6 As we stated in Remark 2, we only need the local information $L(G_i)$ when we design the local supervisor S_i, since S_i achieves the language $inf\overline{CO}_i(T_i(K))$. But to verify the condition in Theorem 1, we still need the global information $L(G)$. This is not the case for the concurrent systems as is obvious from the condition of Theorem 4. So for the concurrent systems, we need not compute $L(G)$. This provides tremendous computational savings for large concurrent systems.

We also have a similar result as Theorem 2 for concurrent systems when each observation mask M_i is identity.

Theorem 5 Let G be a concurrent system with $G =\|_i G_i$, $i \in I$, Σ be the global event set, which is the union of local event sets, *i.e.*, $\Sigma = \cup_i \Sigma_i$, $i \in I$. Let T_i be the natural projection from Σ to Σ_i, Σ_{ci} be the local controllable event set, M_i be the local observation mask. If M_i is identity for each i, then for a given nonempty prefix-closed sublanguage K of $L(G)$, there exist local supervisors $\{S_i,\ i \in I\}$ such that $\cap_i T_i^{-1}[L(S_i/G_i)] = K$ if and only if K is $(L(G), \Sigma_u)$-controllable and $(G, \{T_i, i \in I\})$ co-normal.

Remark 7 Since now $L(G) = \cap_i T_i^{-1}[L(G_i)]$, the co-normality is the same as the separability defined in [10, Definition 2.1]. In [10, Theorem 4.1], the authors studied a similar problem as studied in Theorem 5:

Given a prefix-closed language $E \subseteq \Sigma^*$, let $K = sup\overline{C}(L(G) \cap E)$, find the local supervisors $\{S_i,\ i \in I\}$ such that $\cap_i T_i^{-1}[L(S_i/G_i)] = K$.

Under the assumption that $(\forall i \neq j)\Sigma_{ui} \cap \Sigma_j = \emptyset$, the authors gave the same necessary and sufficient condition as Theorem 5 for the existence of the local supervisors. But in Theorem 5, we do not need the above assumption. So our result extends that of [10].

For concurrent systems, results similar to those of Theorem 3 and Corollary 2 can also be obtained, which we state below:

Theorem 6 Let G be a concurrent system with $G = ||_i\ G_i,\ i \in I$, Σ be the global event set, which is the union of local event sets, *i.e.*, $\Sigma = \cup_i \Sigma_i,\ i \in I$. Let T_i be the natural projection from Σ to Σ_i, Σ_{ci} be the local controllable event set, M_i be the local observation mask. Given two languages A and E with $A \subseteq E \subseteq L(G)$, there exist local supervisors $\{S_i, i \in I\}$ such that $A \subseteq \cap_i T_i^{-1}[L(S_i/G_i)] \subseteq E$ if and only if

$$\cap_i T_i^{-1}[inf\overline{PCO}_i(T_i(A))] \subseteq E.$$

Corollary 3 Let G be a concurrent system with $G = ||_i\ G_i,\ i \in I$, $A = \cap_i T_i^{-1} A_i$ and $E = \cap_i T_i^{-1} E_i$ with $A_i \subseteq E_i \subseteq L(G_i)$. If $inf\overline{PCO}_i(A_i) \subseteq E_i$ for each $i \in I$, then there exist local supervisors $\{S_i, i \in I\}$ such that $A \subseteq \cap_i T_i^{-1}[L(S_i/G_i)] \subseteq E$.

4 Example

In this section, we apply our results to a simple manufacturing system which is taken from [7] and modified. The system, shown in Figure 2, consists of two machines

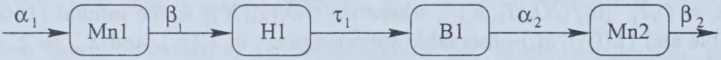


Figure 2: A manufacturing system

$\{Mn_1,\ Mn_2\}$, one material handling device H_1, and one buffer B_1. The capacity of B_1 is 2. We also assume that there are infinite number of workpieces at the input port of Mn_1, and a workpiece is taken away from Mn_2 after the completion of the processing on Mn_2.

The event labels in Figure 2 represents the following actions:

α_1 : a workpiece is taken by Mn_1 from the input port;

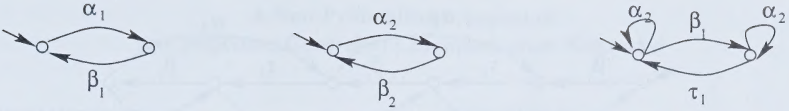
α_2 : a workpiece is taken by Mn_2 from B_1;

β_1 : a workpiece is taken by H_1 from Mn_1 after the completion of its processing;

β_2 : a workpiece is taken away from Mn_2 after the completion of its processing;

τ_1 : a workpiece is sent to B_1 by H_1.

The event τ_1 is the only uncontrollable event. At first, we consider the case that all events are observable. The uncontrolled system is given as $G = G_1 \parallel G_2 \parallel G_3$, where G_1 and G_2 represent Mn_1 and Mn_2 respectively, and are shown in Figure 3(a) and 2(b), G_3 is the subsystem of the combination of H_1 and B_1, and is shown in Figure 3(c). Here we consider only closed languages, therefore all states are considered to be marked states.



(a) (b) (c)

Figure 3: Subsystems G_1, G_2, and G_3

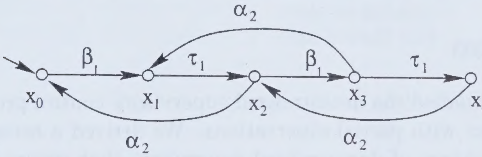


Figure 4: Automaton for A_3

The global event set is $\Sigma = \{\alpha_1, \alpha_2, \beta_1, \beta_2, \tau_1\}$, and the local event sets are $\Sigma_1 = \{\alpha_1, \beta_1\}$, $\Sigma_2 = \{\alpha_2, \beta_2\}$, and $\Sigma_3 = \{\beta_1, \tau_1, \alpha_2\}$. We have $\Sigma_{1c} = \Sigma_1$, $\Sigma_{2c} = \Sigma_2$, $\Sigma_{3c} = \{\beta_1, \alpha_2\}$, $T_i : \Sigma \to \Sigma_i$, $M_i = Id$, for $i = 1, 2, 3$. The range specifications are given as $A = \cap_i T_i^{-1}(A_i)$ and $E = \cap_i T_i^{-1}(E_i)$, where $E_1 = A_1 = L(G_1)$, $E_2 = A_2 = L(G_2)$. A_3 is generated by the automaton in Figure 4, requiring that α_2 be enabled when B_1 is not empty, and β_1 be enabled when B_1 is not full. E_3 is generated by the automaton in Figure 5, requiring that the overflow and the underflow of B_1 be prevented.

It is easy to check that $inf\overline{PCO}_3(A_3) = A_3 \subseteq E_3$, along with the fact that

$$inf\overline{PCO}_i(A_i) = A_i = E_i,\ i = 1, 2,$$

so from Corollary 3 we know that there exist local supervisors $\{S_i, i = 1, 2, 3\}$ such that $A \subseteq \cap_i T_i^{-1}[L(S_i/G_i)] \subseteq E$. In fact, $\{S_i, i = 1, 2, 3\}$ can be synthesized as follows: S_1 and S_2 enable all events at all states; $S_3 = (R_3, \psi_3)$, R_3 is the automaton in Figure 4, ψ_3 disables α_2 at states $\{x_0, x_1\}$ and β_1 at state x_4.

Now suppose that, owing to a failure in a sensor, event α_2 becomes unobservable in G_3, *i.e.*, $M_3(\alpha_2) = \epsilon$. Then we can calculate $inf\overline{PCO}_3(A_3)$, which is the language generated by the automaton in Figure 6. From Figure 5 and Figure 6, it can be seen that $inf\overline{PCO}_3(A_3) \supset E_3$. So the condition in Corollary 3 is not satisfied. Since this condition is not necessary for the existence of the local supervisors, we would still like to know whether we can somehow verify the existence of local supervisors for this case. We can check the condition of Theorem 6. It is easy to verify that for each i, $T_i(A) = A_i$, and so $inf\overline{PCO}_i[T_i(A)] = inf\overline{PCO}_i(A_i)$. Further, we can also show that $\cap_i T_i^{-1}[inf\overline{PCO}_i(A_i)] \supset E$ (the details are omitted here). So the condition in Theorem 6 does not hold, which implies that no local supervisors exist for this case.

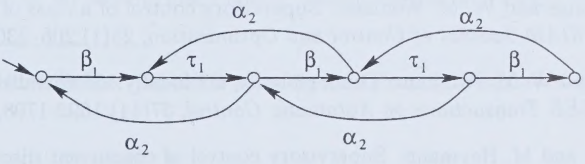


Figure 5: Automaton for E_3

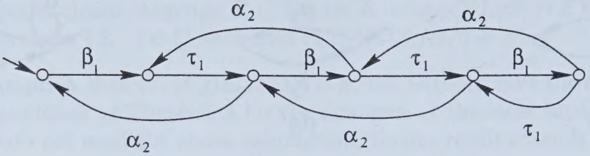


Figure 6: Automaton for $inf\overline{PCO}_3(A_3)$

5 Conclusion

In this paper we studied the decentralized supervisory control problem of concurrent discrete event systems with partial observations. We derived a necessary and sufficient condition for the existence of decentralized supervisors that ensure that the controlled behavior of the system lies in a given range. The ideas were illustrated by application to a simple manufacturing system. This paper extended the results of [1, 9, 6, 7, 10] on the decentralized control of concurrent systems under partial observations.

References

[1] R. Cieslak, C. Desclaux, A. Fawaz, and P. Varaiya. Supervisory control of discrete event processes with partial observation. *IEEE Transactions on Automatic Control*, 33(3):249–260, 1988.

[2] R. Kumar and V. K. Garg. *Modeling and Control of Logical Discrete Event Systems.* Kluwer Academic Publishers, Boston, MA, 1995.

[3] R. Kumar and M. A. Shayman. Formulae relating controllability, observability, and co-observability. *Automatica*, 34(2):211–215, 1998.

[4] S. Lafortune and E. Chen. On the infimal closed and controllable superlanguage of a given language. *IEEE Transactions on Automatic Control*, 35(4):398–404, 1990.

[5] F. Lin and W. M. Wonham. On observability of discrete-event systems. *Information Sci.*, 44(3):173-198, 1988.

[6] F. Lin and W. M. Wonham. Decentralized supervisory control of discrete-event systems. *Information Sci.*, 44(3):199-224, 1988.

[7] F. Lin and W. M. Wonham. Decentralized control and coordination of discrete-event systems with partial observation. *IEEE Transactions on Automatic Control*, 35(12):1330-1337, 1990.

[8] P. J. Ramadge and W. M. Wonham. Supervisory control of a class of discrete event processes. *SIAM Journal of Control and Optimization*, 25(1):206–230, 1987.

[9] K. Rudie and W. M. Wonham. Think globally, act locally: decentralized supervisory control. *IEEE Transactions on Automatic Control*, 37(11):1692-1708, 1992.

[10] Y. Willner and M. Heymann. Supervisory control of concurrent discrete-event systems. *Int. J. Control*, 54(5):1143-1169, 1991.

A New Probabilistic Approach to Congestion Control in Communication Networks[1]

Harold Mortazavian
Computer Science Department, UCLA
4532J Boelter Hall
Los Angeles, CA 90095-1594
E-mail: mor@cs.ucla.edu
Tel: 310 794 4980
Fax: 310 825 5057

Jelena Mirkovic
Computer Science Department, UCLA
4276 Boelter Hall
Los Angeles, CA 90095-1594
E-mail: sunshine@cs.ucla.edu
Tel: 310 825 4033
Fax: 310 825 5057

Abstract

A new congestion control algorithm is proposed in which routers are viewed as *local controllers,* that act upon a distinct subset of events that they can control, on the basis of the partial information locally available. The objective of control is to *alter the probabilistic dynamics* of the network in such a way, that certain specified parameters remain within acceptable bounds (while perhaps extremizing other parameters).

Introduction

The objective of this paper is to present a new probabilistic approach to *congestion avoidance* and *control* in communication networks. We regard a communication network as a probabilistic dynamical system, and routers at various nodes, as local *controllers*. The approach presented is based on a theory of analysis and control of a new class of probabilistic discrete systems developed in [3, 4]. The class of systems considered are characterized by the fact that their behavior consists of a sequence of symbols, abstractly thought of as representing state transitions of an underlying process, under *partial observation*.

With regard to the detection and control of congestion in networks there are two basic approaches: window control and flow control. In packet-switched networks the most commonly used scheme is window control as described in [2]. This approach, however, as several studies have shown [5], leads to oscillatory behavior in the network. This oscillatory behavior has several causes and various undesirable consequences, leading to loss of throughput at the gateways and low utilization of the links. In connection-oriented networks (such as ATM) the most commonly used scheme is flow control. Routers make estimates of the amount of resources available downstream and send only as many packets as the network can accept. This leads to restrictive sending policies and underutilization of the network.

Our approach is aimed at overcoming the deficiencies in both schemes by avoiding restrictive window or flow control and responding to congestion by utilizing alternative paths to a destination. The only assumption we make is that the congestion level can be characterized by a parameter α, $\alpha \in [0,1]$. We assume that a node i, sending packets to a node k, when encountering

---

[1] Work supported by the Defense Advanced Research Projects Agency (DARPA) grant N66001-98-8509.

congestion along a given path u_{ik}, can estimate the level of congestion along the path by a parameter α, but in general has *no* information as to the location of the congestion, namely the set of links on which the congestion has occurred. Further we assume that we have *partial information* about congestion obtained from the information on congestion levels on incident links.

The main idea embodied in our approach consists in: (i) rerouting traffic on alternative paths to their destination, in the face of congestion, instead of controlling sending rates, and (ii) adjusting sending rates at the source, if rerouting does not reduce congestion to an acceptable level. This we achieve under the assumptions of partial control and partial observation, within a systematic theoretical framework. The objective of control is to gradually move traffic away from congested links to parts of the network that have an acceptable level of congestion.

The key concepts are those of probabilistic controllability with a parameter λ, and probabilistic observability with a parameter ω, of a random tree, where λ and ω are lower bounds to the conditional probability of events of interest from the point of view of control and observation and serve as measures of the effectiveness of the control and the observation process. In [3] the relation between the total performance parameter $\mu \in [0,1[$, and λ and ω is obtained in the regular case, as well as in the asymptotic case (namely as the size of the tree goes to infinity). The parameters λ and ω are in turn related to traffic characteristics and the network topology. The dynamics of each router is governed by a differential equation whose solution is the probability distribution of routing traffic along incident links. The parameters λ and ω explicitly appear in this differential equation. Simulation results show significant gain in throughput compared with the single-path routing algorithm.

Main Features of the Proposed Algorithm

The main features of the proposed algorithm, which we shall later describe in detail, are as follows:

1. Each node measures periodically the congestion level on all of its outgoing links for traffic to each destination.
2. Each node calculates the controllability and observability parameters λ and ω (that we shall define), which characterize network conditions in some important respects, from the point of view of control.
3. The parameters λ and ω are defined by the traffic characteristics of the network, and the network topology.
4. Based on this analysis, each node acts autonomously as a local controller, adjusting and updating the sending probability distributions over its incident links.
5. This update process is governed by a differential equation that relates the parameters λ and ω, with probability distribution of sending over the links.

Network, Observation and Control

We assume a Markovian directed network structure with the set of nodes Q and the set of links X. We assume the transitions of this chain are labeled by symbols from the finite set X.

Further we assume that distinct transitions leaving any given state have distinct labels. Thus, we have a discrete probabilistic system

$$A=(X, Q, q_0, P)$$

where X is finite, Q is (at most) countable, and $q_0 \in Q$ is an initial state. The transitions of A are defined by the probability measure:

$$P= Q \times X \times Q \rightarrow [0, 1]$$

satisfying for all $q, q' \in Q$ and $x \in X$:

$$\sum_{(x,q')} P(q,x,q')=1 \text{ and } (\forall x,q \in X \times Q) Card(\{q' \in Q | P(q,x,q') \geq 0\}) \leq 1.$$

X^* is defined as set of all sequences made out of elements of X, including the empty sequence. The language generated by A is defined by:

$$L(A) = \{u \in X^* | (\exists \in Q) P(q_0, x, q') \geq 0\}$$

and denoting a set of final states by $Q_f \subset Q$, by

$$L(A,\lambda) = \{u \in X^* \mid (\exists q' \in Q_f) P(q_0,x,q') \geq \lambda\}, \lambda \in [0,1[.$$

When a *(source, destination)* pair *(i, k)* is specified, *i* is interpreted as the initial state q_0, *k* as the final state Q_f and *L(A)* as the set of all paths from *i* to *k*.

In order to characterize the *congestion control* problem with *partial information* on the location of congestion, along a path, from a source to a destination, we introduce a mapping:

$$\varphi : (X^*, \circ, \varepsilon) \rightarrow (\{x \in \Re \mid x \in [0,1]\}, \max, 0)$$

defined by:

$$\varphi(u) = \max_{x \in alph(u)} \alpha(x), \quad a \in [0,1], \text{ where } alph(u) = \{x \in X | |u|_x \geq 1\}$$

where $|u|_B$ denotes the number of symbols of u that belong to a subset B of the alphabet X. (Note that if $|u|$ denotes the length of u, $|u| = \sum_{x \in X} |u|_x$). Clearly, for all paths $u, v \in X^*$,

$$\varphi(uv) = \max(\varphi(u), \varphi(v)); \varphi(\varepsilon) = 0 \quad , \varepsilon \text{ the null sequence.}$$

so that φ is a homomorphism.

To introduce control and observation we assume some transitions (X_{uc}) are *uncontrollable*, in the sense that their probability is left to the free dynamics of the process, whereas others (X_c) are *controllable* in the sense that their probability can be *determined* by a controller: $X_c = X - X_{uc}$. Further we assume that some transitions (X_{uo}) are *unobservable*, whereas others (X_o) are *observable* $X_o = X - X_{uo}$. (Here, we are building upon the deterministic notions in this area, leading to a probabilistic formulation first presented in [4]).

We introduce a *probabilistic controller* $\mathbb{C}$ defined by the mapping:

$$C : \varphi(L(A)) \rightarrow \boldsymbol{P}_X \text{, where } \boldsymbol{P}_X = \left\{ P_{Y \subset X} : \begin{cases} P_Y = P\,|_Y, & Y = X_{uc} \\ P_Y = P_Y^c, & Y = X_c \end{cases} \right\}$$

subject to a suitable normalization. Here, P denotes the probability measure of A, and P^c denotes the probability measure determined for each $u \in \varphi(L(A))$, over X_C, by the controller *C*, and $\boldsymbol{P}_X$ is the set of all probability measures that can be defined on *X* such that *P* is *invariant* over X_{uc}, under the action of the mapping $C : P_{X_{uc}} = P\,|_{X_{uc}}, P \in \boldsymbol{P}_X$ [3].

Before we proceed further we shall introduce some notation which we shall use in the sequel, and make some further remarks as to our assumptions.

For a node *i* we denote by *I(i)* the set of incident (neighboring) nodes to *i*. We denote by x_{ij} the direct link between *i* and *j*, and by $u_{ij}(k)$ a path from *i* to *k* via *j* (note: there can be many such paths). We denote by $P_{ij}(k)$ the probability that node *i* routes its traffic to *k* via the incident node *j*. The parameter $\alpha_{ij}(k) \in [0, 1]$, denotes the congestion level measured for traffic from *i* to *k* via *j*. (According to this notation, clearly node *i* measures the level of parameter α for each destination *k* and for each incident link leading to this destination). We assume that $P_{ij}(k)=0$ in all cases in which *k* is not physically reachable via *j*. We let $\alpha_r \in [0, 1]$ denote the "reference" congestion parameter value, and if $\alpha_{ij}(k) > \alpha_r$ we say that traffic from *i* to *k* via *j* is experiencing congestion. As mentioned previously, the set of all paths from *i* to *k* via *j* define a *tree* which we shall denote by $T_{ij}(k)$ (This is the case since we approach the problem from the point of view of the source).

The Observability Parameter: ω

Let X be a finite, nonempty alphabet, Y a nonempty subset of X^*, and φ: $X^* \rightarrow X^*$ a morphism. We say φ *observes* Y with parameter $\omega \in [0, 1[$, if for all *u* in Y and *x* in X,

$$P(vx \in Y | ux \in Y, v \in \varphi^{-1}\varphi(u) \cap Y) > \omega\ .$$

We then say that Y is *observable* by the *morphism* φ *with parameter* ω [3]. In the case of trees with the mapping φ as defined previously in this paper, we say analogously that a tree $T_{ij}(k)$ is *probabilistically observable* in *L(A)*, by a mapping φ: $X^* \rightarrow \Re^+$, with parameter $\omega_{ij}(k) \in [0, 1[$, if for all paths *u* and *v* in X^* and *x* in *X*,

(1) $$P(vx \in T_{ij}(k) | ux \in T_{ij}(k), v \in \varphi^{-1}\varphi(u) \cap T_{ij}(k), vx \in L(A)) > \omega_{ij}(k)$$

Thus, we say a tree is probabilistically observable with a parameter ω, if whenever two paths have the same image under the action of the mapping φ (and thus appear the same to an observer) and one when followed by a certain symbol belongs to the tree, the other when followed by the same symbol also belongs to the tree, with probability greater than ω.

It is clear that, if a tree is probabilistically observable with a parameter ω, then when a controller confronts any two paths that appear the same to him, with probability greater that ω his observation is adequate for his decision process of admitting or rejecting next transitions, or distributing probabilities over the set of possible transitions on that node. Note that all paths $u \in T_{ij}(k)$ form an equivalence class that is characterized by the parameter $\alpha_{ij}(k)$, since φ depends on α..

We shall now describe how the observability parameter ω depends on traffic parameters. Denote by $\alpha_{ij}{}^{*}(k)$ the maximum level of parameter α that node i can encounter for traffic to k via j:

$$\alpha_{ij}{}^{*}(k) = \max\left\{\alpha_{ij}(k), \beta_{jl}{}^{*}(k) \middle| l \in I(j)\right\}$$

where

$$\beta_{jl}(k) = \frac{Card\left(alph(u_{jk}) \cap alph(u_{lk})\right)}{Card\left(alph(u_{lk})\right)} * \alpha_{il}(k)$$

where $alph(u) = \left\{x \in X \middle| |u|_x \geq 1\right\}$ as defined previously.

Thus $\beta_{il}(k)$ is a measure of the influence that traffic on the paths leading from i to k via other links (different from x_{ij}) might have on our parameter α.

We shall elaborate on this point further. Since we have partial observation on α, we can not identify the location of the congestion, and since all nodes downstream are performing probabilistic routing, we can rely on $\alpha_{ij}(k)$ being the upper bound on congestion only on the link x_{ij} (since the probability of traversing this link for the whole traffic from i to k via j is 1). However we must estimate the upper bound on α downstream from j. This bound depends also on measured α on other incident links (different from x_{ij}), if the trees defined by these links have branches in common with the tree defined by x_{ij}. The fraction:

$$\frac{Card\left(alph(u_{jk}) \cap alph(u_{lk})\right)}{Card\left(alph(u_{lk})\right)}$$

expresses the potential influence of the congestion level observed on link x_{jl} on the traffic related to the tree defined by x_{ij} (i.e. the tree $T_{ij}(k)$), and the parameter $\beta_{il}(k)$ is a measure of congestion incurred this way.

It follows from the consideration of (1) that $\omega_{ij}(k)$ is the lower bound on

$$P(vx \in T_{ij}(k) | ux \in T_{ij}(k), v \in \varphi^{-1}\varphi(u) \cap T_{ij}(k), vx \in L(A)) .$$

Thus $\omega_{ij}(k)$ is determined by the highest level of α that we can expect to encounter on the path from i to k via j. And we write: $\omega_{ij}(k) = 1 - \alpha_{ij}{}^{*}(k) - \varepsilon$ with ε being an infinitesimal quantity.

Controllability Parameter: λ

We say a tree $K \subset L(A)$ is *controllable* in $L(A)$, on the set X_{uc} with parameter $\lambda \in [0,1[$, if, for each u in X^{*} and x in X,

$$P(ux \in K | u \in K, x \in X_{uc}, ux \in L(A)) > \lambda, \quad \lambda \in [0,1[$$

This is a key concept on which we shall now elaborate further.

We consider the case of a node i sending to a destination k via the node j. As defined earlier, the set of all paths from i to k via j forms a tree denoted by $T_{ij}(k)$. (Namely, a nonempty connected graph with no cycles). We further define as "preferred" or "desirable" (and with some abuse of language as "optimal") a subset of links belonging to $T_{ij}(k)$ on which the level of α is below a reference level α_r. (This can be defined as a *forest* since it may consist of two or more unconnected trees. However, to simplify the terminology, when there is no serious concern for confusion, we shall call such a collection *a preferred tree*). The objective of control is to keep the flow of traffic on this preffered tree. However, on a path of traffic from i to k, some uncontrollable events might occur that can increase the level of the parameter α on some links of $T_{ij}(k)$ thus taking us out of the preferred tree. Such potential uncontrollable events include:

(i) hardware or software failures on a link;

(ii) a node belonging to a path $u_{ij}(k)$ generating traffic on some links $\{ x_{mn} \in alph(u_{ij}(k))\}$;

(iii) a node sharing some links on the path $u_{ij}(k)$ generating traffic over these links.

Thus we relax our definition of "staying on a desired tree" and allow violations, so long as we do not stay out of the tree for a period longer than a certain τ, τ>0.

We say that a (parameterized) subtree $T_{ij}(k, \alpha_r, \tau)$ of a tree $T_{ij}(k)$ from a node i to a node k via j is controllable with parameter λ, λ∈[0, 1[, if given that we are on a path with measured level of α not exceeding α_r, and we encounter an uncontrollable event, with probability greater than λ, after a certain time τ, the path taken will have a measured parameter α not exceeding α_r, (This means that we allow α to exceed α_r, temporarily, during a transition state, when adaptation is taking place, for a maximum time τ).

We formalize the above as follows:

A tree $T_{ij}(k, \alpha_r, \tau)$ from i to k via j is *controllable* in $T_{ij}(k)$, on X_{uc} with *parameters* λ, λ∈[0, 1[, and τ, τ>0, if for each $u \in X^*$, and $x \in X_{uc}$ and for all time t, t≥0:

$$(2)\ \ P(ux(t+\tau) \in \mathrm{T}_{ij}(\mathrm{k}, \alpha_r, \tau) \mid \mathrm{u(t)} \in \mathrm{T}_{ij}(\mathrm{k}, \alpha_r, \tau), \mathrm{x(t)} \in X_{uc}, ux(t) \in \mathrm{T}_{ij}(\mathrm{k})) > \lambda_{ij}(k), \quad \lambda_{ij}(k) \in [0,1[$$

The parameter λ defined in the above is the lower bound on the probability that once we are on a desired tree, we remain on it, after the occurrence of an uncontrollable event.

Let us now examine on what the parameter λ depends and how it should be defined.

(i) Let us suppose that we have an arbitrary, fixed probability of encountering uncontrollable events on $T_{ij}(k)$, and examine what effect the size of preferred tree has on λ. The size of $T_{ij}(k, \alpha_r, \tau)$ is defined by $\alpha_{ij}(k)$. If $\alpha_{ij}(k) << \alpha_r$, then all links on the subtree defined by this link have a sufficiently small level of α, so that the whole $T_{ij}(k)$ (except maybe for few branches) forms our desired tree $T_{ij}(k, \alpha_r, \tau)$ and potentially troublesome parts of the tree are few, and easily avoidable, even in the face of the occurrence of an uncontrollable event. If, however, $\alpha_{ij}(k) > \alpha_r$, there is a large part of $T_{ij}(k)$ that has links with congestion greater than α_r. Thus, our preferred tree $T_{ij}(k, \alpha_r, \tau)$ is sufficiently small, so that the occurrence of any uncontrollable event might take us out of this tree. It follows that λ depends on the observed value $\alpha_{ij}(k)$ and we define it as follows:

$$\lambda_\alpha = \begin{cases} 1-\varepsilon & \text{if } \alpha < \alpha_r \\ \dfrac{1-\varepsilon}{1+(\alpha-\alpha_r)} & \text{if } \alpha \geq \alpha_r \end{cases}$$

(ii) If an uncontrollable event occurs on our path, then the part of the path from the location of the event to k, is unsuitable from our point of view (either it is physically inaccessible or it has an unacceptable level of α). The objective of control is now to try to avoid this part of the path, and we assume that this can be achieved within a period τ, given that there is an alternate path from node directly upstream from the location of the occurrence of an uncontrollable event, to k. Thus defined, λ_{uc} depends on the topology of the network and we define it as follows:

$$\lambda_{uc} = 1 - \varepsilon - \frac{Card\left(\left\{x_{lm} \in T_{ij(k)} \middle| \left(\forall u_{jk} \in T_{ij(k)}\right)\left(x_{lm} \in alph(u_{ij(k)})\right)\right\}\right)}{Card\left(\left\{x \in T_{jl(k)} \middle| \left(x \in alph(u_{jk}), l \in I(j)\right)\right\}\right)}$$

Since λ_α and λ_{uc} are independent, we define λ as: $\lambda=\lambda_\alpha.\lambda_{uc}$

The Differential Equation Governing the Dynamics of a Router

Various theoretical and practical considerations lead us to the following differential equation governing the behavior of each router as a controller, in determining the probability distribution of sending over its controllable incident links:

$$\frac{dP(t)}{dt} = -K \frac{\omega(t)}{1-(1-\lambda(t))\rho} \frac{d\omega(t)}{dt}$$

which yields the update equation upon integration, where K is an empirical constant and ρ is a constant, which we call the *risk tolerance factor*. We note the natural dependence of the updated probabilities on their values in the previous time interval, as well as on the value of the controllability and observability parameters λ and ω, which in turn depend on traffic parameters and network topology. (We remark in passing that while λ and ω are lower bounds to (conditional) probabilities that, given sufficient information can be calculated in a particular case, in general we are really more interested in an approximation to their value as a function of traffic characteristics, as postulated here).

We thus arrive at the following new algorithm for congestion control in communication networks.

The algorithm

Each node performs periodic updates (with period T, T>0), of probability distributions of sending over incident links, per destination, in the following manner:

1. It measures the level of congestion during the previous time interval T.

2. If this level is below α_r on all incident links, the probability 1 is assigned to the link leading to the "optimal path" (e.g. the shortest path), and 0 to all other links. So in the case when there is no congestion, the performance of the algorithm should be the same as that of the classical routing algorithm that favors paths according to the length or the corresponding delay.

3. If on some, but not all links, the level of α is equal to or above α_r, node adjusts probabilities of sending over these links according to the above differential equation, where to simplify the notation, indices i,j and k in λ, ω and P have been omitted. The "probability mass" corresponding to the difference:

$$\Delta P_i{}^k = \sum_{\left\{ j \in I(i) | \alpha_{ij}{}^k \ge \alpha_r \right\}} P_{ij}{}^k((n+1)T) - P_{ij}{}^k(nT)$$

is redistributed over the links that have α below α_r, according to a set of weights, in such a way that links with higher weights are favored:

$$P_{ij}{}^k((n+1)T) = P_{ij}{}^k(nT) - \frac{\Delta P_i{}^k w_{ij}{}^k}{\sum\limits_{\{l \in I(i) | \alpha_{il}{}^k < \alpha_r\}} w_{il}{}^k} \qquad \text{for } \forall j, \alpha_{ij}{}^k < \alpha_r$$

Weights are calculated according to some prespecified criteria (e.g. path length, delay, quality of service, etc.)

4. If on all links the measured level of α is above α_r, node retains the existing probabilities and attempts to alleviate congestion by adjusting traffic parameters (e.g. decreasing sending rates).

Simulation

Simulator was written in C++ langauge. It was designed with the purpose to test our routing algorithm. It implements three routing algorithms.

- Single-path routing algorithm that favors minimum-distance paths. This should simulate OSPF routing.
- Multipath routing algorithm with uniform probability of routing along all links - traffic todestination is routed on all available paths (except those that would create cycles) with equal probability.
- Routing algorithm proposed in this paper

Each of these three algorithms can have two modes of control window adjustment in the case of congestion:

- Slow-start control window - each packet that is acknowledged increases window size by one. Once the congestion is detected by a packet being dropped, window size is reduced exponentially, until congestion ceases. To avoid oscillation and fast building of congestion, linear increase of sending window is applied, until it reaches certain threshold value.
- No reaction control window - control window is incremented by one attach packet acknowledgement. There is no reaction to congestion, so that after some time all sources are sending at their maximum rate

In addition to this, the routing algorithm suggested in this paper does perform some adjustments to control the window size even if option 2 is selected, as described in section2. Each source generates in each time slot a probability distribution for sending to any node. If this probability exceeds some user-defined number, the source will send a random number of packets to the destination. This number can not be larger than *maxcwnd* (maximum allowed size of control window, a constant specified by the user) Within a single time unit the following actions occur:

- Each source generates per destination a probability distribution for sending and also a random number of packets to be sent. If this number is less than the current control window size, that many packets are sent, otherwise the number of packets sent is determined by the control window size. This traffic is propagated to destinations, by allowing packets to move one hop forward in each time unit. Link delays are also accounted for by scheduling packets in the output queues of intermediate nodes. Along the way some packets are dropped and this is recorded. Destinations receive packets and send acknowledgements that are propagated to the source immediately.
- At the end of each iteration statistics are gathered and the observability and controlability parameters are calculated. Then the probability distribution is adjusted according to the proposed algorithm.

Input to the Simulator

The user provides a connectivity file that cntains the information on the topology of the network. The format of lines in the file is:

```
node1 node2 bandwidth-of-the-link buffer-size link-delay
```

The user also specifies the desired simulation scheme (TCP/uniform/probabilistic routing) and the level of tolerance for congestion (in the case of the probabilistic scheme) and the duration of the simulation.

Output of the Simulator

The output of the program is in the file `stat.txt`. For each node it provides the statistics of the portion of packets that were sent into the network and were acknowledged. The overall throughput for the whole network , as well as the overall number of packets dropped are also measured. Additional information is provided in the file `out.txt` – e.g. routing tables of the nodes and probabilities at certain moments in time. This is used mostly for debugging.

Simulation Results

We have tested the proposed routing algorithm using the network topologies depicted in figure 1.

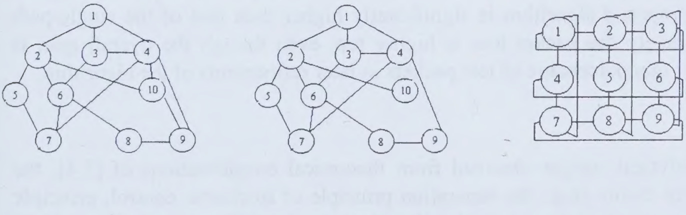


Figure 1: Tested schemes

Simulation results are given in figure 2.

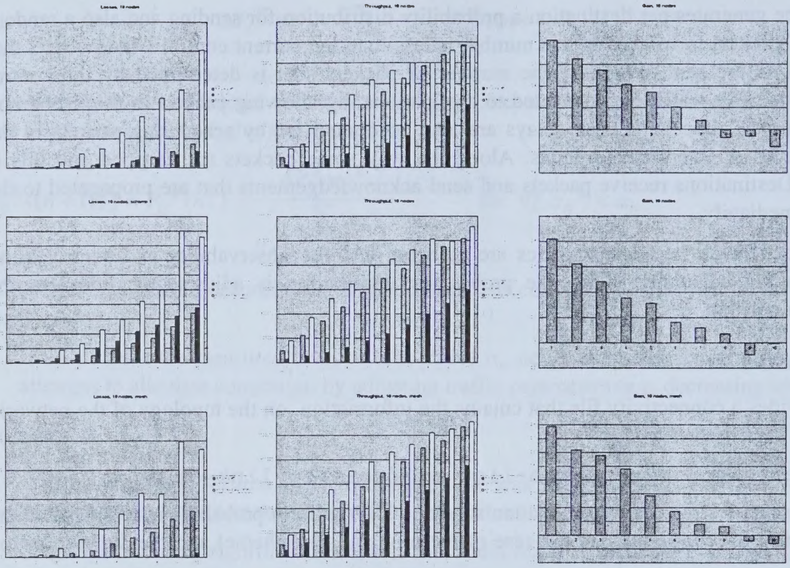


Figure 2: Simulation results

Cases of ten congestion levels were simulated. The leftmost graph shows the information on the total number of packets lost during the simulation. The central graph shows the information on the total number of packets acknowledged. The rightmost graph shows the information on the actual gain that is calculated according to the following formula:

$$Gain=((Throughput_{PA}-Throughput_{SP})-(Loss_{PA}-Loss_{SP}))/Throughput_{SP}$$

where PA denotes the proposed algorithm, and SP the single-path routing algorithm.

As it can be seen from the graphs, in all cases our algorithm performs better than the uniform probability multipath routing algorithm, and thus we exclude this from further consideration. In the cases of low/medium congestion, when there is a chance to solve the problem by rerouting, the overall throughput of the proposed algorithm is significantly higher than that of the single-path routing algorithm. Unfortunately the packet loss is higher too, even though the overall gain is positive. We expect to lower this percentage of lost packets in later refinements of the algorithm.

Conclusion

On the basis of analytical insight obtained from theoretical considerations of [3,4], the classical principles of control theory (e.g., the separation principle of stochastic control, principle of optimality of dynamic programming,...) and the knowledge gained from methods currently applied in the field, we have derived a congestion avoidance and control algorithm. The key concepts are those of probabilistic controllability and observability of trees, the conception of routers as local controllers that act on a distinguished subset of contollable events using the partial information available locally, and the differential equation governing the behavior of each router as a local controller. Much remains to be done, as regards theory and applications.

References

[1] D. Bertsekas, R. Gallager, *Data Networks*, Prentice Hall, 1992.

[2] V. Jacobson, M.J.Karels, "Congestion avoidance and control", Proceedings of Sigcom'88, pp 1-21.

[3] H. Mortazavian, "A theory of analysis and control of a class of discrete probabilistic systems". Submitted for publication.

[4] H. Mortazavian, "On control of stochastic discrete event systems", Proceedings of the 2nd European Control Conference, Groningen, The Netherlands, June 28 - July 1, pp 2191-2196, 1993.

[5] L.Zhang, S.Schenker, D.D. Clark, "Observations on the dynamics of a congestion control algorithm: the effects of two-way traffic", Proceedings of Sigcomm'91, pp 1-25.

[6] D. Bertsekas, "Dynamic behavior of shortest path routing algorithms for communication networks", IEEE Transactions on Automatic Control, 1982, pp. 60-74.

[7] F. Lin, W.M. Wonham, "On observability of discrete event systems", Information Sciences 44(3), 1988, pp.173-198.

[8] P.J. Ramage, W.M. Wonham, "Supervisory control of a class of discrete event processes", SIAM J. Control and Optimization 25(1), 1987, pp. 206-230.

A Bigraph Matching Theorem [*]

Sami Ayyorgun Rene L. Cruz
Dept. of Electrical and Computer Engineering
University of California at San Diego
La Jolla, CA 92093-0407 USA.
ayyorgun@cwc.ucsd.edu cruz@ece.ucsd.edu

Sep. 22, 1999

Abstract

We pose two *matching* problems on *graphs*, and solve a special case of one of them. That case is realized on *bigraphs* by considering a partition of the each part of a bigraph, and a mapping of this partition of its nodes into the positive integers. We are interested in matchings such that the number of nodes matched in any part of the partition does not exceed the integer that it is mapped into. We then give a theorem showing the cardinality of a *maximum* matching in such problems, and suggest an efficient(polynomial time) algorithm to find one such maximum matching.

In this paper, we pose two *matching* problems on *graphs*, and solve a special case of one of them. Before we go ahead and do so, let us give a very brief terminology and notation—first for sets, and then for graphs—as needed for mathematical clarity.

We denote the intersection of sets A and B simply as AB. The set difference of A from B is denoted as $A - B$. A *cover* of a set A is a set $\{A_1, A_2, \dots\}$ of non-empty sets A_i called *parts* whose union includes A, and said to be *proper* if the union is A. A proper cover is called a *partition*, if its parts are mutually disjoint. Lastly, $|A|$ denotes the *cardinality* of a set A.

Let V be a finite set, and E be a set of its subsets with two elements. Then, V and E are together called a *simple* and *undirected* **graph** $G(V, E)$. The elements of V are called the *nodes*, and those of E are called the *edges*. The two nodes composing an edge are called its *end-nodes*, and said to be *joined* by the edge. A graph is *simple* if the end-nodes of neither one nor two of its edges are the same, and *undirected* if none of its edges is an ordered pair. In this study, graphs are understood to be finite, simple, and undirected. A graph is called a **bigraph** if its nodes could be partitioned into two sets, say M and W called its parts, such that all edges have one end-node in M and the other in W. In this case, the bigraph is denote as $G(M \cup W, E)$. A **matching** in a graph is a set of edges where no two of which have a common end-node, and is said to be *maximum* if no matching in the graph has a larger cardinality. A node is said to be *matched* in a matching, or sometimes just 'matched' if the matching is inherent, if it is an end-node of an edge in the matching, and *unmatched* otherwise. Finally, an edge is said to be *included* in a matching, if it is an element of the matching, and *excluded* otherwise.

---

[*] This research was supported in part by the National Science Foundation under grant NCR-97-25568 and by the Center for Wireless Communication at UCSD.

We now pose both of the problems. Let us cast only one of them, and then get the other one easily from that by replacing some words.

> (**V-cover-constrained Matching**) Suppose a graph, a proper cover of its nodes, and a mapping of the cover into the positive integers are given. Find the cardinality of a maximum matching such that the number of nodes matched in any part of the cover does not exceed the integer that it is mapped into.

The counterpart of this, **E-cover-constrained Matching**, is obtained simply by replacing the words 'nodes' with 'edges', and 'matched' with 'included'. We refer to these problems in short as "V-cover" and "E-cover" problems, respectively. An interesting special case of the each problem is realized when the cover is a partition. In that case, we call them **V-partition-constrained Matching** and **E-partition-constrained Matching**, which in turn are referred to briefly as "V-partition" and "E-partition" problems, respectively. In these problems, a part of a cover is said to be *unsaturated* by a matching, or sometimes just 'unsaturated' if the matching is inherent, if the number of its elements matched or included in the matching is less than the integer that it is mapped into, and *saturated* otherwise.

Next, let us consider a special case of V-partition problem on bigraphs. It is induced by a partition of the each part of a bigraph $G(M \cup W, E)$—obviously, not the most general V-partition problem on bigraphs—and a mapping f of the partition, say $\{V_i : i \in S\}$ where each part is clearly a subset of either M or W, into the positive integers. We could assume without loss of generality that there is no *isolated* node(nodes which are not an end-node of an edge) in G, since they neither affect a matching nor truly alter a constraint. Also, since the solution of the problem is bound to the matchings agreeing to the constraints, only those are referred to be a 'matching' in G, unless otherwise stated.

Now, let us give a few definitions after which a solution for this case will follow as a theorem. The first two are general ones. Consider a graph $G(V, E)$; the **image** $\Gamma(A)$ of a set of nodes A is the set of all the nodes to which the nodes in A are joined in G, and the **principal** $\Pi(A)$ of a set of nodes A is the set of all the nodes whose image is a subset of A. The next is a particular one for this problem. A function σ is defined on the power set of $M \cup W$ as follows

$$\sigma(A) = \sum_{i \in S} \min\{f(V_i), |AV_i|\}.$$

Theorem *The cardinality of a maximum matching in G is* $\min\limits_{A \subseteq W}\{\sigma(M - \Pi(A)) + \sigma(A)\}$.

proof: Consider a maximum matching μ in G. Let A be a subset of W, and x be the set of edges in μ with an end-node in A. Since μ is a matching, $\sigma(A)$ and $\sigma(M - \Pi(A))$ could not be less than $|x|$ and $|\mu - x|$, respectively. So, since A is arbitrary, all the terms over which the min is taken in the theorem are greater than or equal to $|\mu|$. Hence, the min $\geqslant |\mu|$. Suppose, the min $> |\mu|$. Then, as it could be seen by the terms in the min corresponding to $\emptyset$ and W, at least two parts, one is a subset of M and the other is of W, are not saturated by μ. In other words, at least two nodes, one in M and the other in W, are unmatched and in unsaturated parts. Let X_0 and Y_0 be all such nodes in M and W, respectively. Define $\mu(A)$, $P(A)$, and $Q(A)$ with respect to μ, for a set of nodes A, to be the set of all the nodes matched in μ to some node in A, the set of all the matched nodes p such that $\Gamma(A)$ has an unmatched node in the part to which p belongs, and the set of all the unmatched nodes q such that $\mu(A)$ has a node in the part to which q belongs, respectively. Consider the following recurrence for $k \geqslant 0$

$$\begin{aligned} Y_{k+1} &= \Gamma(X_k) \cup P(X_k) \\ X_{k+1} &= (\mu(Y_{k+1}) \cup Q(Y_{k+1})) - \bigcup_{i \leqslant k} X_i. \end{aligned}$$

The premise on which the recurrence is constructed is the fact that μ is a maximum matching. This implies that $Y_0\Gamma(X_k)$ must be $\emptyset$ for any k, otherwise a matching with cardinality $|\mu| + 1$ would be guaranteed by the recurrence as follows. Let l be the first k for which $Y_0\Gamma(X_k) \neq \emptyset$. Pick any node p_l in $Y_0\Gamma(X_l)$, and let q_l be any node in X_l joined to p_l. If q_l is matched in μ, then a_l is just another tag for it. Else, let a_l be any matched node in X_l and in the part to which q_l belongs. Let b_l be the node matched in μ to a_l. Pick any unmatched node p_{l-1}, if any, in Y_{l-1} and in the part to which b_l belongs. Else, p_{l-1} is just another tag for b_l. Apply the same arguments to p_{l-1} and to all the rest of p_i's until q_0 is tagged. At the end, a matching with cardinality $|\mu| + 1$ would be reached by including all the edges $\{p_i, q_i\}$ in μ, and removing all the edges $\{a_i, b_i\}$ from μ. Finally, let $X = \bigcup_i X_i$ and $Y = \bigcup_{i>0} Y_i$. Notice that $X \subseteq \Pi(Y)$, $\sigma(Y) = |\mu(X)|$, and $\sigma(M - \Pi(Y)) = |\mu| - |\mu(X)|$. So, the term in the min corresponding to Y is equal to $|\mu|$, which is in contradiction to the assumption(the min $> |\mu|$). That concludes the proof.

Note that also suggested in the proof explicitly, is an efficient(polynomial time) algorithm to find a maximum matching in G. We leave the details of such an algorithm to some later work.

There are also other formulations of this solution, one of which we give next as a corollary whose proof is essentially the same as of the theorem. Needless to say, M and W are also interchangeable in both the theorem and the corollary, due to the symmetry in bigraphs.

Corollary *The cardinality of a maximum matching in G is* $\min_{A \subseteq M}\{\sigma(A) + \sigma(W - \Gamma(A))\}$.

There are—of course—other proofs of this theorem, too. A somewhat "higher layer" proof—thus, provoking further understanding—, exploits the celebrated **max-flow-min-cut Theorem** [1].[1] The proof is immediate once the flow network is constructed around G as follows. Introduce the source and sink nodes s and t, respectively, and a node v_i for each part V_i. Join s to the nodes v_i for which V_i is a subset of M, and let the capacity of the edge $\{s, v_i\}$ to be $f(V_i)$. Do the same thing once more, but replace s with t, and M with W. Join every node v_i to the nodes in V_i, and let their capacity to be 1. Finally, let the capacity of any edge in E to be "very large", say ∞. Clearly, a *maximum s–t flow* in this construction corresponds to a maximum matching in G.

As a last remark, we note that König's Minimax Theorem [1], where there is essentially no partition constraint, is a special case of the theorem presented in here.

Reference

[1] L. Lovasz, M.D. Plummer. *Matching Theory.* North-Holland, Amsterdam, New York, 1986.

---

[1] This was pointed out by Prof. Fan C. Graham, Dept. of Mathematics at the University of California, San Diego.

Chunks in PLAN:
Language Support for Programs as Packets

Jonathan T. Moore Michael Hicks Scott Nettles
Department of Computer and Information Science
University of Pennsylvania
200 South 33rd Street
Philadelphia, PA 19103
{jonm,mwh,nettles}@dsl.cis.upenn.edu*

Abstract

Chunks are a programming construct in PLAN, the Packet Language for Active Networks, comprised of a code segment and a suspended function call. In PLAN, chunks provide support for encapsulation and other packet programming techniques. This paper begins by explaining the semantics and implementation of chunks. We proceed, using several PLAN source code examples, to demonstrate the usefulness of chunks for micro-protocols, asynchronous adaptation, and as units of authentication granularity.

1 Introduction

The current IP-based Internet has been a success in part due to the simplicity of its architecture. However, this success, while bringing greatly increased use, has also greatly increased the demand for new and more complex network services and protocols. The goal of Active Networking is to allow these demands to be met by increasing the flexibility with which the network can be changed. Active networking achieves its increased flexibility by making the network programmable.

In the SwitchWare project [2], we have been exploring how to make the network programmable both by allowing switches to be dynamically extended with new services and by allowing packets themselves to be programs. A key way in which we have explored the idea of packets as programs is through the design and implementation of PLAN, the Packet Language for Active Networks, a domain specific language for writing packet programs [8]. Using PLAN we have implemented PLANet [9], an internetwork in which every packet is a PLAN program, and in which all activities in the network proceed by evaluating PLAN packets. This paper draws heavily from examples developed while building PLANet.

Just because packets are programs does not mean that many of the familiar features of conventional packets do not need to be supported. In particular, the desire to build networks using layering requires that PLAN programs support encapsulation, while the need to support services such as checksumming and fragmentation means that PLAN programs must sometimes be treated as data. In this paper, we explore the key mechanism used in PLAN to support functionality like encapsulation and

---

*This work was supported by DARPA under Contract #N66001-96-C-852, and by the National Science Foundation CAREER Grant #CCR-9702107, with additional support from the Intel Corporation.

treating packets as data, a language construct called *chunks.*

Chunks consist of some code, a named function used as an entry point, and arguments for that function. When chunks are evaluated, they invoke the named function using the arguments. By transporting a chunk from one node to another before evaluating it, we achieve a kind of remote evaluation; such remote evaluation how PLAN supports the distributed communication and programming from which the network is built. Chunks are first-class data values, which means they can be used as function arguments (notably, as an argument in another chunk), returned from functions, stored in data-structures, and manipulated as bags of bits.

If chucks merely provided an exotic way of doing what can already be done they would be of limited interest. However, chunks, and the programs embodied in them, provide flexibility that goes beyond traditional packet headers. Besides showing how chunks support traditional networking functionality, we also show how chunks can be used in three additional broad ways: to support the implementation of *micro-protocols* [11], to provide *adaptive protocols* [6, 14], and to solve certain security problems in a novel way.

To provide context, we begin by presenting a brief background of PLAN and chunks in Section 2. Next, we discuss and provide examples of the applicability of chunks to micro-protocols, adaptive protocols, and security in Sections 3–5. Finally, we discuss related work in Section 6 and conclude with future research directions in Section 7.

2 Background

Our examples use PLAN and so here we provide a brief look at its features, focusing on chunks. The reader is referred to [8] for a more in-depth treatment.

PLAN is a simple, lexically-scoped language designed to be carried in packets and to fill and extend the role of traditional packet headers. To this end, PLAN's expressiveness is limited to permit active nodes to evaluate a PLAN program without requiring authentication: PLAN programs are functional in style and are guaranteed to terminate. Its syntax resembles that of ML.

To mitigate their limited expressive power, PLAN programs may invoke node-resident *services* that provide access to information (e.g., time of day, number of network interfaces, etc.) or to more powerful functionality (e.g., fragmentation, reliability, soft state, etc.). These services may be dynamically loaded over the network and may be written in a more standard high-level language like C, Java, or ML.

Every PLAN program contains a top-level "code hunk," or *chunk.* As mentioned, chunks have three components: PLAN code, an entry point function within that code, and bindings for the function's parameters. A chunk can be viewed as a snapshot of a traditional function call where the arguments have been assembled and the program is about to branch to the function itself. When a PLAN packet arrives at its destination node, its entry point function is called with the given arguments.

The most powerful aspect of chunks, however, is that they are first-class values in PLAN. The PLAN syntax

$$|f|(expr_1, ..., expr_n)$$

is an expression of type chunk which creates a new chunk. Intuitively, the |'s indicate the part of the expression whose evaluation is to be delayed: the function call itself. Specifically, the created chunk consists of the same code as the current execution, an entry point f, and bindings which are the values obtained by evaluating $expr_1, ..., expr_n$. Thus, chunks can be manipulated, copied, or passed as arguments, and, most notably, can even appear as bindings in other chunks. Indeed, encapsulating one chunk within another is a key mechanism we will use in the rest of the paper.

Besides treating it as data, the main action that can be performed on a chunk is to execute it: the eval service evaluates its chunk argument by loading the code segment and invoking the entry point function on its arguments. If we take the view, as above, that a chunk is a suspended function call, then `eval` resumes the computation, but with of course a different continuation.

To support such functionality as fragmenting or checksuming a chunk, we also provide the ability to create a concrete representation of a chunk in the form of a byte string. This representation is essentially the marshaled wire-format version of the chunk. A service related to `eval`, `evalBlob`, serves to evaluate this representation. Unlike the uses of code as data in Lisp-like languages, we so far we have found that uninterpretable strings of bytes suffice for our concrete representation.

3 Micro-protocols

Most commonly-used protocols like TCP and IP are complex, with a variety of functionality and many options. Developing and testing such protocols can be difficult and error prone, and the resulting protocols are not particularly flexible. These problems have motivated past research on micro-protocols [13, 14]. Each micro-protocol embodies a single function or option; more complex behavior is achieved by composing these micro-protocols.

We will present here two micro-protocols: one for a packet checksum, and one for fragmentation. These protocols will also serve to show how chunks provide some basic networking implementation techniques within the context of packets as programs.

In PLAN, micro-protocols are composed through chunk encapsulation. In general, each micro-protocol takes a chunk plus additional arguments and returns one or more new chunks which add the micro-protocol's functionality. This is similar to encapsulation in traditional networking, where, as a packet moves down the network stack, each protocol layer encapsulates the higher-level packet while perhaps adding additional header information for itself.

```
svc verifyChecksum : (blob,int) -> bool
svc evalBlob : blob -> 'a

fun unchecksum(c:blob, sum:int): unit =
  if verifyChecksum(c,sum) then
    (evalBlob(c);())
  else
    () (* drop packet *)
```

Figure 1: Code for a checksum chunk

```
svc reassemble :
  (blob,int,bool,key) -> 'a

fun defrag(frag:blob, seqnum:int,
                morefrags:bool, session:key)
           : unit =
  (reassemble(frag, seqnum, morefrags,
                session); ())
```

Figure 2: Code for a fragmentation chunk

For example, suppose we had a chunk *c* to which we would like to add checksumming. We can invoke a `checksum` service on *c* which converts it into a stream of bits (i.e., a "blob" type in PLAN) via the standard PLAN marshaling system, computes a checksum *sum*, and then wraps them in a chunk with a code segment like that shown in Figure 1. When this new chunk *d* is evaluated, `unchecksum` is called using *c* and *sum* as arguments. `Unchecksum` then calls the `verifyChecksum` service to ensure that *c* still has the proper checksum, and then either evaluates *c* or aborts, as appropriate.

As another example, consider the task of fragmentation; namely, we have some chunk *c* to transmit and evaluate remotely, but it may be larger than the MTU of the intervening path. We can have a `fragment` service which takes *c* (and the MTU size), represents it as a "blob," and divides it into MTU-sized[1] pieces. Each fragment is then wrapped in a chunk with a code segment as shown in Figure 2. The bindings for

---

[1] Less the overhead for the reassembly chunk.

```
svc defaultRoute : host -> host * dev

fun send_frag (x:int*host,c:chunk)
               : int * host =
  (OnRemote(c,snd x,fst x,defaultRoute);
   x)

svc checksum : chunk -> chunk
svc fragment : (chunk,int) -> chunk list
svc getMTU : dev -> int
svc length : 'a list -> int
svc getRB : void -> int

fun udp_deliver (b:blob, p:port,
                 dest:host) : unit =
  let val c:chunk = |deliver|(p,b)
      val d:chunk = checksum(c)
      val p:host*dev = defaultRoute(dest)
      val ds:chunk list =
             fragment(d,getMTU(snd p))
      val l:int = length(ds) in
   (foldl(send_frag,
          (getRB()/l,dest),
          ds); ())
  end
```

Figure 3: UDP-style delivery

these new chunks would each have a piece of the original chunk, a sequence number, an indication whether it was the last fragment or not, and a unique identifier. The new chunks, when evaluated, simply register themselves with the `reassemble` service on the destination. This service collects all the incoming fragments, puts them in the proper order, reconstitutes the original chunk, and then evaluates it.

Figure 3 shows the composition of our two protocols to form a UDP-like delivery service. At the highest level, we have some data (represented as a `blob`) which we want to deliver to a specific port on a specific host. First we create the chunk *c*, which encapsulates this behavior. Then we create a new chunk *d* which adds checksumming. After querying the appropriate path MTU, we then invoke the `fragment` service to get a list of fragmentation chunks. If the original chunk was small enough to fit within one link-layer frame, these chunks take the form shown in Figure 4[2]. Finally, we use `foldl` to apply `send_frag` to send each fragment to *dest*.

When the fragments arrive they will be evaluated causing them to be placed in the reassembly table. Once all the fragments have arrived, they will be reassembled into chunk *d*, which when evaluated will verify the checksum, resulting in chunk *c*. When *c* is evaluated it will call `deliver` with port argument *p* causing data argument *d* to be delivered to the correct port.

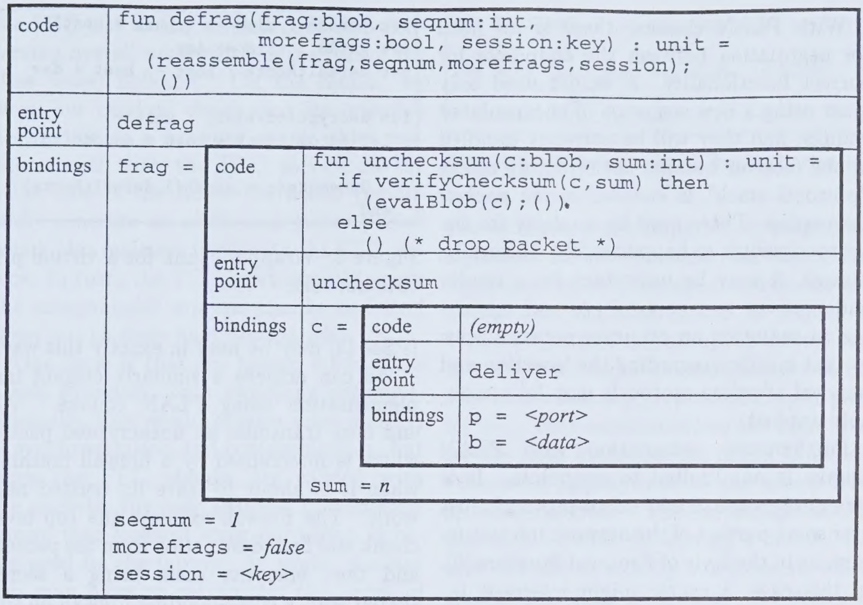
Both of the above micro-protocols share the same basic structure: a service on the source is invoked with a chunk plus some configuration parameters. This results in the creation of a new chunk which carries the code to perform the destination side of the micro-protocol; note that this code may refer to services that reside on the destination (but need not necessarily reside on the source). The new chunk could then potentially be wrapped in yet another micro-protocol or simply sent across the network. At the destination, the chunks simply "unwrap" themselves.

This common structure makes many things easy. For one, we can have policy drive the composition of micro-protocols, rather than having dependencies built into complex protocols. Indeed, in our above example, rather than have fragments of a checksummed delivery packet, we could have invoked *fragment* first, and then done a `checksum` on each resulting chunk, thus ending up with checksummed fragments of the original chunk. Each micro-protocol takes a chunk and returns a chunk or list of chunks, so they may be arbitrarily ordered in a type-correct way. Of course, the order *does* matter from a semantic point of view.

Secondly, micro-protocols may be coded to remove redundant functionality. For ex-

---

[2] If fragmentation was actually required, we would have only a part of the innermost two chunks; however, for ease of illustration, we have not shown this case.



```
fun defrag(frag:blob, seqnum:int,
                morefrags:bool, session:key) : unit =
  (reassemble(frag,seqnum,morefrags,session);
   ())
```

```
fun unchecksum(c:blob, sum:int) : unit =
  if verifyChecksum(c,sum) then
    (evalBlob(c);()).
  else
    () (* drop packet *)
```

Figure 4: Chunk encapsulation

ample, if a path only has Ethernet interfaces, the checksum service might simply return the original packet, as the checksum would be redundant with the underlying CRC check. Similarly, the fragment service can (and does) just return the original chunk if it was already small enough. Either of these optimizations remove the need to execute certain receiving code at the destination.

In fact, the destination will not even have to do a test to determine that it need not execute the receiving code! Since the demultiplexing path is encoded in the the way the chunks are encapsulated, the unnecessary code will simply not be called as an arriving chunk "unwraps" itself. This mechanism is in fact quite powerful and allows us to do straightforward asynchronous protocol adaptation, as we see in the next section.

4 Asynchronous Adaptation

Adaptive protocols are ones that can be dynamically reconfigured. In particular, they can react to changing network conditions to improve performance. For example, if a data stream is bottlenecked due to a low bandwidth link, it might be desirable to compress the stream. Similarly, many checksum errors arising from a noisy link might suggest using some sort of error correction scheme to introduce redundancy.

In most approaches to adaptive protocols, a primary problem is synchronization. Namely, a source and a destination must agree on the structure of the protocol stack they are using: a protocol where the sender encrypts data but the receiver fails to decrypt it would hardly be useful. As described in [14], such signaling protocols can often be complex (and sometimes expensive).

With PLAN chunks, there is no need for negotiation between the endpoints for correct functionality. A sender need only start using a new sequence of encapsulated chunks, and they will be correctly handled at the receiver because the structure of the "protocol stack" is *encoded in the packets themselves.* There need be no delay for the protocol switch to happen safely. Naturally, though, it may be important for a sender and receiver to communicate and cooperate to maintain an accurate network view so that a policy regarding the insertion and removal of micro-protocols may be reasonably applied.

Furthermore, adaptation with PLAN chunks is not limited to endpoints. It is straightforward to add micro-protocols just over some portion of the network infrastructure, as in the style of Protocol Boosters [6]. In this case, a router might intercept incoming PLAN packets, wrap their top-level chunks in a new "boosting" micro-protocol, and send them on to a "de-boosting" location. Once there, the wrapper chunk will perform the receive-side of the micro-protocol and then send the original top-level chunk on to its final destination. Conventional, non-active packets can be treated the same way, essentially letting them tunnel in an active packet to allow dynamic protocol composition.

We make these ideas more concrete by presenting two examples. Both involve micro-protocols which are applied at points within the network rather than just at the endpoints of a communication.

Virtual private networks. Our first scenario is that of virtual private networks, where we have several networks of trusted nodes that we wish to connect by traversing untrusted links. We would like to give all the end hosts the illusion of being within a single trusted network. This can be accomplished in a straightforward manner by encrypting and encapsulating packets between trusted networks. In the Internet, IPSec [4] may be used in exactly this way.

```
svc decode : blob -> (chunk * host)
svc getRB : void -> int
svc defaultRoute : host -> host * dev

fun decrypt(cd:blob) : unit =
  let (c,d):chunk * host = decode(cd)
  in
    OnRemote(c,d,getRB(),defaultRoute)
  end
```

Figure 5: Wrapper chunk for a virtual private network

We can achieve a similarly elegant implementation using PLAN chunks. An end host transmits an unencrypted packet which is intercepted by a firewall machine when it is about to leave its trusted network. The firewall extracts the top level chunk and final destination from the packet, and then encrypts them using a secret shared with a corresponding firewall on the other side of the untrusted network. The firewall then creates a new wrapper chunk like that shown in Figure 5: the binding `cd` is the bit string resulting from the encryption.

When the wrapper arrives at the opposite firewall, it calls the `decode` service which accesses the firewall's shared secret to recover the original (unencrypted) chunk *c* and evaluation destination *d*. Finally, it sends *c* along to *d*. This scheme could be easily extended to encrypt additional fields (e.g., a nonce to prevent replay attacks).

Mobile computing. Our second scenario considers mobile computing over wireless links, which are more noisy than wire-based LANs. Wireless links often have poor TCP throughput, as negative acknowledgments due to checksum failures are interpreted (incorrectly) as network congestion. To compensate for packet errors, the networking software on the laptop could engage a forward error correction (FEC) micro-protocol when operating in mobile mode.

With PLAN chunks, we can easily limit

the FEC just to the wireless link, thus conserving overall bandwidth in the rest of the (less lossy) network. On the source, we wrap our original chunk and its intended destination in a wrapper chunk which registers itself with the FEC service on the other side of the link. We would periodically generate an additional parity packet which also registers itself with the FEC service. In turn, the FEC service would verify the encapsulated original chunks and send them out to their final destinations.

One issue is that the laptop might cross a cell boundary, thus switching gateways. Normally, this would require some amount of synchronization and communication, but since the FEC chunks are carried with the packets, the new gateway immediately knows that forward error correction is being used by the laptop. At worst, the laptop may have to retransmit the batch of packets which were being transmitted when the switch occurred. Here, the fact that PLAN packets actively specify their processing saves us additional communication over the lossy link.

5 Security

PLAN was designed as a restricted language with the intention that programs using only services that provide appropriate protection would not require authentication. In fact, other active networking research [1] has shown that authenticating every active packet results in unacceptable performance degradation.

On the other hand, for full advantage to be taken of the flexibility of active networking, we want to allow PLAN programs to use services that require that packets be authorized. Functions about which very few *a priori* claims can be made suggest the need for some sort of authorization policy controlling their installation and use.

PLAN chunks provide an elegant mechanism for this sort of authorization. A user with greater-than-normal privilege could take a chunk and compute a cryptographic signature for it. In turn, the chunk and signature could be encapsulated in another chunk like the one shown in Figure 6. When evaluated, the `authEval` service verifies that the signature matches the chunk and user's identity (represented by a public key), thus *authenticating* the user. The service then evaluates the chunk in an environment appropriate to the user's level of authorization. The advantage of this approach is that in signing a chunk, the user can define exactly the scope in which greater privilege is needed: the duration of the evaluating the chunk. An alternative certificate-based approach is potentially less precise.

```
svc authEval : (chunk,blob,blob) -> 'a

fun wrap (c:chunk, sig:blob, id:blob)
       : unit =
  (authEval(c,sig,id); ())
```

Figure 6: Wrapper chunk for authorization

Of course, this scheme only permits a user to authenticate himself to an active node, but not vice versa. There is also a vulnerability to replay attacks, and public key cryptosystems are notoriously slow. Fortunately, all three of these problems can be addressed, as described by the Secure Active Network Environment (SANE) [3], and implemented in our current testbed, PLANet [7].

6 Related Work

Micro-protocols are used in the *x*-kernel [11, 13]. However, the stress there is on software engineering and code reuse; although their approach enables the relatively simple development of new protocols, they must still be composed statically, whereas micro-protocols implemented with PLAN chunks can be dynamically reordered.

Ensemble [14] is a toolkit for distributed application development. Unlike the *x*-kernel, applications may adapt and dynamically reconfigure their protocol stacks. However, Ensemble uses a Protocol Switch Protocol which halts communication, synchronizes through a central coordinating participant, and then resumes communication. PLAN chunks do not require this pause for synchronization and do not need centralized coordination. Furthermore, PLAN chunks are not limited to an end-host only regime of operation.

Protocol Boosters [6, 12] interpose additional functionality within the network infrastructure. These boosters enhance performance in a way that is transparent to the applications communicating across the "boosted" subnets. However, for multi-component boosters, signaling is required to support the addition or removal of a booster. Finally, because they reside in the network infrastructure itself, some boosters are subject to failures due to routing changes sending boosted packets around their intended de-boosting element. PLAN chunks are not subject to these failures because the chunk encapsulation essentially records which micro-protocols have been applied and must be undone at the destination.

Application-specific safe message handlers (ASHs) [15] also support dynamic protocol layering in the Exokernel [5]. Dynamic code generation is used to allow integrated layer processing based on upcalls. Since PLAN chunks are carried in packets, it is not clear that dynamic code generation or just-in-time compilation will provide an advantage over simple interpretation. Furthermore, the evaluation of PLAN chunks already naturally suggests an upcall-based organization. Lastly, PLAN chunks are not limited to end hosts.

Other active networking projects such as Netscript [17] and ANTS [16] offer flexible networking as well. In particular, Netscript permits dynamic recomposition of protocols, but still requires synchronization across nodes to make sure that protocol stacks match correctly. It is not clear whether ANTS protocols are composable as their programming abstractions ensure that protocols cannot interfere with one another.

7 Future Work and Conclusions

One of the drawbacks of our current implementation is that PLAN evaluation is currently quite costly. Partially this is a result of lack of tuning, and the use of a byte-code interpreted language implementation. A faster PLAN interpreter and service routines might be constructed using certified run-time code generation techniques as in the Cyclone [10] compiler.

A general concern about our approach is the space cost of carrying the code in the packet. To mitigate this overhead, we are currently considering ways in which the participants in a protocol may cache code rather than always transmitting it with the packet. One promising approach is to add language-level *remote-references* that may be thought of as pointers to remote objects. Since all PLAN values (including chunks) are *immutable*, the contents of a remote reference may be safely cached without the need for a coherence protocol. This approach might also allow us to cache some of a chunks bindings as well, which might result in further space savings.

Implementation of various Protocol Boosters should be straightforward using PLAN chunks. Given a more streamlined PLAN infrastructure, it should be possible to experimentally demonstrate the utility of these techniques as well as other adaptive protocols. In particular, it would be intriguing to boost regular TCP/IP traffic across a PLAN-based subnet.

In this paper, we have described chunks, a construct of our domain-specific programming language, PLAN, and have provided several examples of their use. Firstly, chunks representing micro-protocols can be arbitrarily ordered and composed to create more complex protocols. Secondly, since the encapsulated chunks are carried in the packets themselves, they permit asynchronous protocol adaptation. Finally, chunks provide a convenient level of granularity for program authentication. Indeed, chunks are a powerful enough abstraction to permit the elegant expression of a number of powerful networking mechanisms.

References

[1] D. S. Alexander, Kostas G. Anagnostakis, W. A. Arbaugh, A. D. Keromytis, and J. M. Smith. The Price of Safety in an Active Network. Technical Report MS-CIS-99-02, University of Pennsylvania, January 1999.

[2] D. Scott Alexander, William A. Arbaugh, Michael W. Hicks, Pankaj Kakkar, Angelos D. Keromytis, Jonathan T. Moore, Carl A. Gunter, Scott M. Nettles, and Jonathan M. Smith. The SwitchWare Active Network Architecture. *IEEE Network Special Issue on Active and Controllable Networks*, 12(3), May/June 1998.

[3] D. Scott Alexander, William A. Arbaugh, Angelos D. Keromytis, and Jonathan M. Smith. A Secure Active Network Architecture: Realization in SwitchWare. *IEEE Network Special Issue on Active and Controllable Networks*, 12(3):37–45, May/June 1998.

[4] R. Atkinson. Security Architecture for the Internet Protocol. RFC 1825, August 1995.

[5] Dawson R. Engler, M. Frans Kaashoek, and Jr. James O'Toole. Exokernel: An Operating System Architecture for Application-Level Resource Management. In *ACM Symposium on Operating Systems Principles (SOSP)*, 1995.

[6] D. C. Feldmeier, A. J. McAuley, J. M. Smith, D. Bakin, W. S. Marcus, and T. Raleigh. Protocol Boosters. *IEEE JSAC, Special Issue on Protocol Architectures for the 21st Century*, 16(3):437–444, April 1998.

[7] Michael Hicks. PLAN System Security. Technical Report MS-CIS-98-25, University of Pennsylvania, July 1998.

[8] Michael Hicks, Pankaj Kakkar, Jonathan T. Moore, Carl A. Gunter, and Scott Nettles. PLAN: A Packet Language for Active Networks. In *Proceedings of the 1998 ACM SIGPLAN International Conference on Functional Programming (ICFP'98)*, pages 86–93, September 1999.

[9] Michael Hicks, Jonathan T. Moore, D. Scott Alexander, Carl Gunter, and Scott Nettles. PLANet: An Active Internetwork. In *IEEE Conference on Computer Communications (INFOCOM)*, March 1999. To appear.

[10] Luke Hornof and Trevor Jim. Certifying compilation and run-time code generation. In *ACM SIGPLAN Conference on Partial Evaluation and Semantics-Based Program Manipulation (PEPM)*, January 1999.

[11] Norman C. Hutchinson and Larry L. Peterson. The *x*-Kernel: An Architecture for Implementing Network Protocols. *IEEE Transactions on Software Engineering*, 17(1):64–76, January 1991.

[12] A. Mallet, J. D. Chung, and J. M. Smith. Operating Systems Support for Protocol Boosters. In *HIPPARCH Workshop*, June 1997.

[13] Sean W. O'Malley and Larry L. Peterson. A Dynamic Network Architecture. *ACM Transactions on Computer Systems*, 10(2):110–143, May 1992.

[14] Robbert van Renesse, Ken Birman, Mark Hayden, Alexey Vaysburd, and David Karr. Building Adaptive Systems Using Ensemble. Technical Report TR97-1638, Cornell University, July 1997.

[15] Deborah A. Wallach, Dawson R. Engler, and M. Frans Kaashoek. ASHs: Application-Specific Handlers for High-Performance Messaging. In *ACM Communication Architectures, Protocols, and Applications (SIGCOMM'96)*, August 1996.

[16] David J. Wetherall, John Guttag, and David L. Tennenhouse. ANTS: A toolkit for building and dynamically deploying net work protocols. In *IEEE OPENARCH*, April 1998.

[17] Y. Yemini and S. da Silva. Towards programmable networks. In *IFIP/IEEE International Workshop on Distributed Systems: Opera tions and Management*, L'Aquila, Italy, 1996.

On the Interface of Programmable Network Elements

Gísli Hjálmtýsson
AT&T Labs – Research
AT&T Shannon Laboratories, Florham Park, NJ
gisli@research.att.com

Abstract

There is now increasing interest in active and programmable networking. An increasing amount of research is ongoing on topics ranging from dynamic service and network management, to control plane switch programmability, to active network capsules carrying code in every datagram. The primary motivation for a service provider to increase the programmability of network elements is to provide differentiated services using commodity components. While economizing on infrastructure drives the use of commodity network elements, service differentiation is essential to maintain profitability.

In this paper we discuss the programmable interface of the Pronto router. The interface separates the service generic facilities of an otherwise commodity network node from the service specific functions implementing the service model offered by the network operator. These service specific programs might be provided by end-users, by the service provider, or by an independent service developer and do not require intricate knowledge of the implementation of the service generic facilities.

Keywords: Active networking, programmable networking, API, network services

1 Introduction

There is a significant and increasing interest in active and programmable networking. An increasing amount of research is underway on topics ranging from dynamic network and service management, to control plane switch programmability, to active network capsules carrying code in every datagram. Although these approaches differ in the time scales that networks are programmed and on what time scale the user supplied programs operate on, they share the common objective of creating a more flexible network architecture. In part, this objective is driven by the desire to have more service specific functions within the network, while making the network infrastructure more service independent.

From a network service provider's perspective, the primary motivation for programming networks is to provide differentiated services using commodity components. A programmable network architecture promises insulation from implementation choices of different vendors, as well as insulation from differences and changes in underlying technology. In contrast, with current network architectures significant deployment of equipment from one vendor makes it practically impossible (expensive) to use equipment from another. This problem is all the more important as intense competition is causing mergers and alliances of service providers that use very different infrastructures.

Programmable networks simplify service management, and facilitate rapid service creation. A service provider can develop new services exploiting the network level facilities such as queuing, scheduling and multicasting capabilities, in a proprietary manner without the need for prior standardization. Alternatively, a service provider could purchase the service specific software from independent service developers. Separating the service generic network technology from the service specific technology allows service providers to focus on service creation and compete by offering non-commodity services, while benefiting from increasing competition among network element providers.

The focus of this paper is on the service program interface defined as part of our work on the Pronto (*Programmable Networks of Tomorrow*) architecture. The interface separating the service generic functions supplied by the network element vendor from the service specific

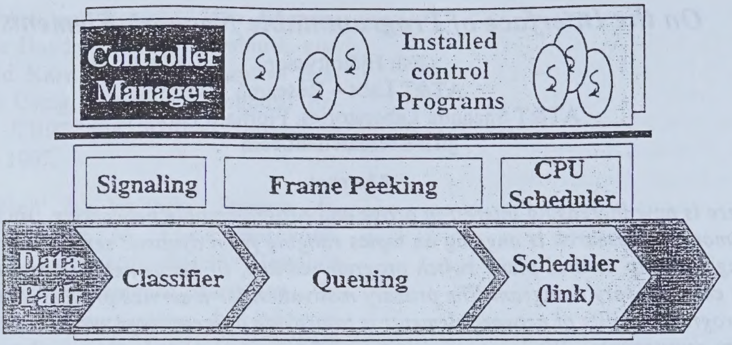


Figure 1 : The Pronto architecture

2 Summary of the architecture and the programmable service models

The Pronto active router architecture (Figure 1) consists of three major parts: a forwarding engine providing service generic facilities, an execution environment where service specific functions reside and execute, and the interface between the two. The most essential part of the architecture is the separation of the service generic facilities of the forwarding engine from the service specific functions running in an execution environment. An execution environment consists of a virtual machine and an environment manager. The environment manager is responsible for authenticating, installing and managing the programs executing in its environment. In addition, the environment manager dispatches (control) messages programs in its environment. Each environment manager has access to an HTTP proxy, and a certificate authority proxy. Typically both are run on the router. Although execution environments can be dynamically installed or removed, we assume these are infrequent operations, and performed with super-user privileges. The only execution environment we have used in the prototyping of the interface described herein is the native virtual machine.

Below the abstraction interface is the forwarding engine and other service generic operating systems and hardware facilities. We have identified six main abstract facilities. In the data path the three major components are a flow classifier, a queue manager and a link scheduler. These facilities currently exist in all routers although they are typically less capable than our architecture allows. More recent higher-end routers support richer classification, the ability to configure multiple queues and simple scheduling among these queues. We push the envelope on each of these components in terms of functionality, and flexibility. In particular the datapath can support fine-grained buffer management and link scheduling among flows.

Outside of the data path, Pronto includes a service generic signaling processor, a resource manager, and a query and response engine (QRE). The first enhances and generalizes current IP signaling to better support service specific signaling. The resource manager schedules access to processors, memory, and ensures resource isolation of the service specific programs. The QRE supports queries and subscription for flow statistics and packet content. An important issue is to minimize data movement across the abstraction boundary. This is achieved using *"frame peeking"* that supports queries (and subscriptions) to a limited number of bytes per packet. After processing the result of a query (or subscription) the service control program may request per packet actions on the packets in the queue, in particular packet discard, changing the drop priority or setting a congestion-indication bit. Part of the ongoing research conducted, as part of this work, is to conservatively enrich the set of QRE primitives and published events, and evaluate their benefits and complexities.

2.1 Summary of the programmable model

The Pronto architecture supports a range of programming models that differ in how they interact with the data path. On one extreme, a Pronto service could operate purely in the control plane, with no interference in the forwarding path. On the other extreme, the service could reside fully in the data path, operating on every data packet. In addition, we have identified two important intermediate cases. In the former, the service exploits the QRE facilities, to support fine time scale control plane functions and frame peeking, to tailor policies based on network conditions and on (part of) the contents or the size of the packets. The second intermediate service model provides a node-local multicast where a packet is forwarded normally, and also delivered to the associated user process. This gives us *four service models*: a pure control plane, finer grained control plane with frame peeking, local multicast, and the last acting fully in data path. These four service models become progressively more capable and costly, with the first two having no impact on data forwarding performance, the third incurring a duplication cost for local delivery, and with the in-data-path processing incurring the execution of a user-level specific program on every datagram.

An important part of the architecture is that for the first three service models, the service specific programs are applied asynchronously from data forwarding. In particular, using the frame peeking and the local multicast service models allow data packets to be forwarded asynchronously to the service processing. Consequently, services employing these service models act as enhancements only and are not essential for forwarding correctness. Due to the enhancement nature it is not essential to process every packet. Clearly the more predictable scheduling of the service specific programs the greater the enhancement. All of the Pronto service models are based on typed flows, rather than individually typed packets. This allows for optimizations and amortization of cost over multiple packets, but also provides an abstraction for services to act at the connectivity level as well as at the data level.

A crucial property of the Pronto architecture is that different service models may be applied to different flows and sub-flows. In particular, one may use a different model for the forward flow than the reverse flow of a particular connection. For example, to facilitate retransmissions of lost packets from a Pronto based proxy to a client, the proxy would trigger the local multicast capabilities to receive a copy of every packet of the forward flow, without interfering with the forwarding path. The proxy would then use the in-data-path model for the reverse flow, intercepting all the ACK or NACK packets from the receiver(s) and use those to trigger the retransmission of lost packets. In another case, a Pronto router would leave the forwarding path untouched except during congestion, where it would selectively discard certain packets (such as the *B* frames in an MPEG stream). Some of the most intriguing research issues in the interface design of the architecture are how and to what extent this type of delegation can be achieved through the interface.

3 The programmable interface

The Pronto interface defines a set of primitives that service specific programs use to interface (and interact) with the underlying forwarding engine. The architecture defines six major components below the interface, a packet classifier, a queue manager, a link scheduler, a CPU scheduler, a control message dispatch module (signaling), and a query and response engine (QRE). We have implemented all of these components and their interfaces. The rest of this section describes the interface to each of these components in detail.

3.1 The packet classifier

The most essential facility of the forwarding engine is the packet classifier, which separates the packet stream into flows. Abstractly, the classifier maps packets into equivalence classes based on attributes of the packet. Typically these attributes include the IP header and the transport level header. The equivalence relation is defined by a filter, F, so that two packets, p_i and p_j, are equivalent, iff $F(p_i) = F(p_j)$. We define the equivalence classes, in terms of a byte mask, M, and a value V, such that $F(p) = (p \& M = V)$, where the & is the bitwise and operator. Multiple filters may map to the same equivalence class, i.e., an equivalence class is defined as $E = \{ p \mid F_0(p) \text{ or } F_1(p) \dots F_n(p) \}$. Each equivalence class has associated with a state, containing in particular the forwarding state. The forwarding state consists of a set of queues and output devices (most often the set is a singleton), or a local socket, or both. Additional attributes may be assigned to an equivalence class, including a request for flow pinning, reservations and more. The most important attribute for network programming however is the program associated with the equivalence class.

The packet classifier supports the following primitive operations:

Add a filter: add(filter_def f, devices d, queues q, socket s) ***return*** *flow_handle;*
- **specify:** a filter definition, and optionally either a list of devices (output ports) and a queue on each device, or a local socket or both.
- **effect:** creates an entry in the forwarding table (forwarding engine's classifier) mapping a filter rule to either the output devices (ports) and the corresponding queues, or a local socket or both. If an output device is specified without a queue the default queue is assigned.

Remove: remove(flow_handle h);
- **specify:** a flow handle.
- **effect:** removes the flow declaration from the forwarding table.

Update: update(flow_handle h, device d, oqueue q, socket s) ***return*** *flow_handle;*
- **specify:** a flow handle, an output device and a queue, and optionally a local socket
- **effect:** modifies the mapping associated with the flow. Note the filter definition does not change. Processing and interpretation otherwise as when creating a flow.

Get: get(flow_handle) ***return*** *{ list of <device, queue> pairs, the local socket }*
- **specify:** a flow handle.

3.2 The queue manager

Queues are manipulated indirectly through the link scheduler, and the query and response engine. Whereas the link scheduler manipulates the queues by scheduling packet transmission, and by assigning queues to the scheduling hierarchy, the content of queues can be manipulated through the query and response engine. The only direct interaction with the queue manager is to create a queue, using the primitive:

Create queue: create_queue(name, device) ***return*** *queue*
- **specify:** name, device.
- **effect:** creates a named queue on the specified device.

3.3 The query and response engine

The query and response engine allows the installed control programs to access the data path asynchronously either by querying or through subscriptions. The interface implements the frame peeking. Subscriptions allow the service specific controllers to monitor progress of flows, for example by subscribing to sequence numbers of a particular flow. The result of a subscription is a flow of fixed length peek segments, one segment for each packet forwarded. Each segment is the result of peeking into the packet as declared in the subscription.

A query gives a limited read access to flow statistics, and to the packets in the queue at the time the query is executed. The result of a query is a signed structure, containing flow statistics, and an array of segments of peek results, one segment for each packet in the queue (Figure 2). A peek result, is simply the bytes copied from each packet (peeked at). The statistics include data transmitted, data queued and data lost, given in bytes and packets.

signature	*statistics*	*Array of peek results*

Figure 2 - The query result structure

After processing a query the service specific program may respond with a signed array of operations, one operation for each packet (Figure 3). The signature of the response is that of the query and allows the forwarding engine to decide if the response is valid. The response engine processes the response by applying the operations to the corresponding packets. As the query is processed asynchronously to data forwarding, some number of packets may be transmitted before the response arrives. To account for this the forwarding engine keeps a count of packets sent since the last query, and ignores the operations for those packets already gone. When the count exceeds a threshold it invalidates the signature (a simple way to realize this is to initialize the counter with the queue length when the query is issued and then decrement the counter on packet transmission down to 0).

signature	*Array of operations, one operation per pakcet*

Figure 3 The response structure. For each packet of the corresponding query, the response specifies an operation. Operations may vary in length.

The three primary primitives supported by the query and response engine are:

*Query: **query**(queue, offset, length, buffer *, buffer_len) **return** bytes_received*

- **specify:** queue, peek offset, peek length in bytes, buffer, buffer size.
- **effect:** fills into buffer (assuming size is big enough) queue statistics, including number of packets/bytes transmitted/lost/in queue. If number of bytes peeked at is > 0, then in addition returns an array with one entry for each packet in the queue, each entry consisting of peek-length bytes, starting from offset. The buffer is then signed for future reference.

*Subscribe: **subscribe**(queue, offset, length, socket)*

- **specify:** queue, peek offset, peek length in bytes, local socket.
- **effect:** creates a subscription for peek-length bytes, starting from offset, of each packet. Subscription results are retrieved by reading from the local socket.

*Response: **response**(queue, response structure)*

- **specify:** queue, response structure, containing a signature, the size of the instruction string, and an array (string) of instructions.
- **effect:** the instructions are supplied in a buffer, whose first entry is the signature from the query, followed by a sequence of (currently fixed length) instructions on packet disposition, one instruction per packet. The current prototype supports three types of instructions: no-op, discard and set congestion indication bit. The forwarding engine maintains a count of packets transmitted since the signature was generated, and thus is able to skip the instructions for packets that have already been transmitted.

3.4 Resource management

To provide performance isolation and support service differentiation we have implemented fine-grained hierarchical start time fair scheduling, both for processing and network transmission. Our scheduler does not provide real time guarantees, but provides cumulatively predictable scheduling (for details see [1]). The schedulers schedule the two resources independently. This is consistent with our enhancement service model, greatly simplifies the scheduling implementation, and provides sufficient assurances for the applications we have considered.

3.4.1 The link scheduler

The link scheduler implements predictable and fair bandwidth allocation. Equivalence classes are mapped to queues and output devices (ports). For each output device a scheduler is created and initialized, creating a root for the scheduling hierarchy and a default FIFO queue. The link scheduler interface supports the following six primitives:

*Initialize scheduler: init_scheduler(device) **return** scheduler;*

- specify: a device name.
- effect: installs the link scheduler on the specified device. Creates a root node in the scheduling hierarchy, and creates a FIFO leaf node for the default queue.

*Get root: get_root(device) **return** handle_to_root*

- specify: device name.

*Add an internal node: add(name, path, weight) **return** internal-node*

- specify: name, path, weight.
- effect: creates a node in the hierarchy specified by path and with the given weight.

*Add a leaf: add_leaf(name, path, weight, service-policy) **return** leaf-node*

- specify: name, path, weight, and a service policy.
- effect: creates a named leaf node at the point in the hierarchy specified by path and using the specified weight, and the queue service policy. The queuing policy is either FIFO (default), or start time fair queuing (SFQ).

*Find node: find(name, path) **return** node*

- specify: node name, and path.

Remove node: remove(node)

- specify: node handle.
- effect: removes the node specified and its queue.

In addition, leaf nodes have two methods for attaching and moving queues. For SFQ leafs, the weight is given as a second parameters. Given a leaf node the primitives are:

Attach: attach(queue, weight)

- specify: queue, and weight for SFQ.
- effect: attaches the queue to the leaf node, with the given weight if SFQ.

Move queue: move(queue, weight)

- specify: queue and weight.
- effect: same as attach (above), but gracefully detaches from the previous leaf node.

3.4.2 The CPU scheduler

The CPU scheduler is realized in an analogous manner as the link scheduler. However, the scheduler is initialized and the root node created as part of the kernel initialization. Apart from the root manipulation, the interface is analogous:

*Add a node: add_node(name, path, weight, service-policy) **return** node*

- specify: name, path, weight, and service policy.
- effect: creates a named node and a new queue at the point in the hierarchy specified by the path, having the specified weight, and employing the specified service policy.

*Find node: find(name, path) **return** node (if found)*

- specify: node name and path.

Remove node: remove(node)

- specify: a node handle.
- effect: remove the specified node and its queue.

An for a leaf node:

Attach and move a process: attach(processor-id), move(processor-id)

- **specify:** processor id and node.
- **effect:** makes node responsible for the scheduling of the process. If moving, gracefully detaches from the prior node.

3.5 Control Message Dispatcher

The control message dispatch interface is used by a signal processor to subscribe to control messages from the forwarding engine. There are two levels of signal processing, one processing service generic messages, the other processing service specific messages. Each environment manager is responsible for dispatching service specific messages to the appropriate control processes. Although in principle service generic messages could be processed by a single entity, for simplicity in our prototype we simply provide a library for this processing that is linked into each environment manager. In our prototyping each environment manager is a user level process, started with super-user privileges and thus able to open a raw socket. The only primitive of the control message dispatch, ***register environment***, is thus essentially setting IP level socket options to this raw socket. The primitive is specified as:

register environment: env_register(socket, protocol_id, alert values (as a vector))

- **specify:** local socket, alerts (and opitonally alert values) accepted given as protocol id.
- **effect:** registers an environment by defining a (raw) socket monitored by the environment manager, and declaring the set of router alerts that the environment manager wants to receive. Note: an environment is started with root privileges.

4 Using the interface

We have implemented numerous applications using the Pronto interface. We discuss two of them below. The first one is stream thinning as congestion adaptation and uses the QRE interface to do selective discard on a media stream, using only the low-end programming models supported by Pronto. Thinning an MPEG stream has become the canonical example of active networking, but all prior discussion puts the stream adaptation fully in the data-path. In the latter example, implementing reliable multicast, the control program exploits the more aggressive program models, using the local copy capabilities for the forward path, but acts fully in the data-path on the reverse flow, demonstrating how the different program models can be applied.

4.1 Stream thinning as congestion adaptation

We have implemented selective discard for and MPEG encoded video stream. MPEG is hierarchically coded using three types of frames I-frames, P-frames and B-frames. Whereas the loss of an I-frame can affect all frames until the next I-frame (a group of pictures), loosing a small number of P- and B- frames only marginally degrades quality. The MPEG selective discard policy discards the less important P- and B-frames in an attempt to protect the to I-frames.

Our approach to congestion adaptation is based on the observation that during congestion buffers are long and thus queuing time significant. Figure 4 shows a pseudo code for the MPEG stream thinning. When run, the video stream controller executes a query (Figure 4, line 4) and blocks. When invoked the request returns flow and queue statistics, and a list of entries for each packet currently in the queue, each entry containing the result of the frame peeking query (in this case the frame type from the packet). Most of the time, the queue is small (or empty) and the controller goes to sleep. If however, the queue exceeds a high water mark, indicating congestion, the controller prepares a vector of operations, one for each packet, discarding all B and P frames until a low watermark is crossed. On completion the

```
1.  Selective_Discard() {
2.    While (1) {
3.      Sleep(some time);
4.      (Qinfo, matches, signature) := query (Q, offset, peekLen);
5.      if (Qinfo.Bytes < highWaterMark) continue;
6.      response r = new response(signature, ||matches|| );
7.      for ( i=0; i < || matches ||; i++ ) {
8.        byte frame_type := matches[i];
9.        r[i] := NOOP;
10.       if (frame_type != I-frame and Qinfo.Bytes > lowWaterMark) {
11.         r[i] := DISCARD;
12.         Qinfo.Bytes -= packet->length;
13.       } // endif;
14.     } // endfor
15.     respond( r );
16.   } // endwhile
17. }
```

Figure 4: Code segment implementing selective discard for MPEG

controller returns this vector with the signature of the query using the respond primitive (Figure 4, line 15).

This example also nicely illustrates the effectiveness of frame peeking for stream thinning. We used this scheme on a short MPEG movie, encoded into 26546 packets, a total of 13.86 MB. Most datagrams are of size 560 bytes, with few of size 170 bytes, with an average of 522 bytes per datagram. Peeking at the full application level header (17 bytes) would have resulted in about 450 KB (3.3%) of the data being copied to user space. Copying only the one byte needed to identify the MPEG frame type (as is the case in Figure 4), reduces this down to 27 KB (0.2%). Enlarging the packets (the MTU in our testbed is 1500 bytes) would lower this ratio further.

4.2 Reliable multicast

Whereas the first example exploits the QRE interface, to implement reliable multicast in an efficient and scalable manner we exploit the ability to act more aggressively in the data-path. There are two main issues in scalable reliable multicast: to delegate retransmission work from the sender to other participants in the multicast, and to limit retransmissions to only those receivers needing the retransmitted packet. We assume losses are indicated by NACKs.

An application specific repair agent (controller) is installed and run on routers in the multicast. For simplicity we assume this is done on all routers of the multicast. (Measurements on the MBONE [2] suggest that repairs need only be offered at a few strategic routers). The session uses multiple multicast groups, say G_0 to G_n. The original transmission is sent on group G_0. Retransmission of a packet whose sequence number mod n is s is sent on group G_{s+1} (see [3] for details). The repair agent uses the local multicast capabilities to receive a copy of all packets on the forward path of the multicast (i.e., opens up a local socket and registers it in the flow state of all $n + 1$ groups). For the reverse path, the agent goes fully into the data-path, intercepting every NACK.

The repair agent maintains a circular buffer of the last k packets received on the forward path. When gaps in the packet sequence occur, it requests a retransmission as if it were a receiver, by sending a NACK upstream. When receiving a NACK for packet i, the repair agent determines if it has the packet, or if it is already awaiting its retransmission. If neither is true, it forwards the NACK upstream. If packet i is locally available, the agent immediately sends it out on $G_{(i \bmod n)+1}$. Otherwise, it adds the incoming interface of the NACK to the multicast group of the missing packet, $G_{(i \bmod n)+1}$, and finally adds the packet (number) to a "*waited-for*" set on that port of that group. When receiving a retransmission on G_j the agent removes the packet from the waited-for set on all output ports of G_j. If this leaves the waited-for set empty for a particular output port, the agent removes that port from the state of G_j.

Using this scheme the retransmission work is delegated to the routers of the multicast thus achieving scalability. Using relatively few groups the receivers receive only small amounts of unwanted retransmissions [3]. Note that the repair agent caches packets for a limited time (as newer packets override the older ones), and forwards NACKs that it cannot serve. Receivers periodically repeat their NACKs for missing packets. Thus the *responsibility* for reliable delivery ultimately remains at the end-systems. The use of multiple multicast groups allows us to exploit the underlying forwarding engine and is thus efficient. Although the reverse path flows completely through the repair agent, the forward stream is forwarded directly by the forwarding engine. Under reasonable loss assumptions, the reverse stream will be a small fraction of the forward stream.

5 Related work

While the number of projects on active and programmable networking [4,5,6,7,8,9,10,11] is rapidly growing, this work mostly draws on the active networking work in IP described in [4,5], and on the work on control plane programmability in ATM networks in [6,7]. The fourth service model of Pronto, acting fully in the data path, has the capabilities of [4]. In fact for in-data-path programming the Pronto architecture would support the model of [5] as an execution environment. However, unlike [4] in where the traditional forwarding model is replaced with a new active one we focus on identifying and separating the service specific functions from the service generic facilities and define the interface between the two. The Pronto interface supports the capabilities of [6,7], but also offers a range of service models.

In the context of Pronto and the interface described herein, [8] describes programmability below the networking layer (protocol update of an active bridge) and [9] programmability of the networking layer (adapting telephony's advanced intelligent networking (AIN) [12] to packet networks). The Darwin project [10] explores similar issues as pursued in the Pronto project but with strong emphasis on resource management in virtual networks. The work in [11] on router plug-ins describes an interesting approach to update kernel level router software. As such it represents programmability below the Pronto interface.

6 Conclusion

In this paper we have described the Pronto programmable interface. The interface separates the service generic facilities of an otherwise commodity network node from the service specific functions. These service specific programs might be provided by end-users, by the network provider, by an independent service developer or by the network element manufacturer. The essential characteristic is that the functions running above the interface do not require intricate knowledge of the implementation of the service generic facilities. All aspects of the interface described have been prototyped on our prototype software router.

A key strength of the Pronto architecture is that it supports multiple service models: strict control plane programmability, finer grained control plane with frame peeking, local multicast, and the last acting fully in data path. This allows service developers to trade off functionality for performance as appropriate. Moreover, different service models may be applied in different parts of the network depending on the ratio of processing capabilities to bandwidth capacity. Service models can be selected on a per (sub-)flow basis.

We have discussed two applications as an example of how the interface can be used to solve interesting problems: a selective discard for media streaming congestion adaptation, and active recovery for reliable multicast. Although active networking solutions have previously been proposed for these applications, using the Pronto interface yields a practical and comparatively lightweight solution by exploiting the appropriate service model supported by the Pronto architecture.

The Pronto architecture defines a network element node OS support. The interface is equivalent to and could be realized as additional system calls in a node OS. The interface and the service generic facilities are applicable for network elements (routers/switches), gateways, proxies and QoS servers. Similarly, the interface applies across environments, including IPv4, IPv6, MPLS and ATM.

7 References

[1] Pawan Goyal and Harrick M. Vin, "Start-time Fair Queuing: A Scheduling Algorithm for Integrated Services Packet Switching Networks (Extended Version)," Accepted for publication in *IEEE/ACM Transactions on Networking.*

[2] M. Yajnik, J. Kurose, and D. Towsley, ``Packet Loss Correlation in the MBone Multicast Network," *IEEE Global Internet Conference*, London, Nov. 1996.

[3] S. Kasera, G. Hjálmtýsson, D. Towsley, and J. Kurose, "Scalable Reliable Multicast Using Multiple Multicast Channels," submitted to *IEEE/ACM Transactions on Networking.*

[4] D. L. Tennenhouse and D. J. Wetherall, "Towards an Active Network Architecture," *Computer Communication Review*, 1996.

[5] D. Wetherall, J. Guttag, and D. L. Tennenhouse, "ANTS: A Toolkit for Building and Dynamically Deploying Network Protocols," *IEEE OPENARCH'98*, San Francisco, April 1998.

[6] Biswas, J., Huard, J.-F., Lazar, A.A., Lim, K.-S., Mahjoub, S., Pau, L.-F., Suzuki, M., Torstensson, S., Wang, W. and Weinstein, S., "The IEEE P1520 Standards Initiative for Programmable Network Interfaces," *IEEE Communications Magazine*, Vol. 36, No. 10, Oct. 1998, pp. 64-70.

[7] J. van der Merwe, S. Rooney and I. Leslie and Simon Crosby, "The Tempest - A practical framework for Network Programmability," in *IEEE Network*, 1998, vol. 12, no. 3, pp. 20-28.

[8] D. Scott Alexander, Marianne Shaw, Scott M. Nettles and Jonathan M. Smith, "Active Bridging," *Sigcomm 1997*, Cannes, France, September 1997.

[9] Samrat Bhattacharjee, Ken Calvert and Ellen W. Zegura. "An Architecture for Active Networking," *High Performance Networking (HPN'97)*, White Plains, NY, April 1997.

[10] Prashant Chandra, Allan Fisher, Corey Kosak, T. S. Eugene Ng, Peter Steenkiste, Eduardo Takahashi, Hui Zhang, "Darwin: Resource Management for Value-Added Customizable Network Services," *Sixth IEEE International Conference on Network Protocols* (ICNP'98), Austin, Oct. 1998.

[11] Decasper, D., Dittia, Z., Parulkar, G., and Plattner, B., "Router Plugins - A Modular and Extensible Software Framework for Modern High Performance Integrated Services Routers," in *Proceedings of ACM SIGCOMM'98*, September 1998

[12] Bell Communication Research, Inc., "AIN Release 1: Service Logic Program Framework Generic Reqirements," FA-NWT-001132.

Bowman and CANEs: Implementation of an Active Network*

S. Merugu[†] S. Bhattacharjee[‡] Y. Chae[†] M. Sanders[†] K. Calvert [‡] E. Zegura [†]

[†]College of Computing
Georgia Tech
Atlanta, GA

[‡]Dept. of Computer Science
Univ. of Maryland
College Park, MD

[‡]Dept. of Computer Science
Univ. of Kentucky
Lexington, KY

Abstract

In parallel with active networks research, efforts have been underway to define and standardize an architectural framework. The framework divides the functionality of an active network node into two components: an Execution Environment (EE) defines a programming interface that allows users to control the active network, a NodeOS defines a set of basic functions to access and manage the resources of the active node. EEs use the abstractions provided by the NodeOS to build a virtual machine made available to users. Feedback from implementation efforts is critical for refining the standards being developed.

We are building an active network comprised of the CANEs execution environment and the Bowman NodeOS. Bowman is constructed by layering active-network-specific operating system functionality on top of a standard host operating system. The host operating system provides low level mechanisms; Bowman provides a channel communication abstraction, an a-flow computation abstraction and a state-store memory abstraction, along with an extension mechanism to enrich the functionality. The CANEs EE provides a composition framework for active services based on customizing a generic underlying program by injecting code to run in specific points called slots. This paper reports on our experience in implementing Bowman, instantiating CANEs on top of Bowman, and developing applications within CANEs.

1 Introduction

In parallel with research experience in active networks, efforts have been underway to define and standardize an architectural framework [3, 2]. In the draft framework, the functionality of an active network node is divided between Execution Environments (EEs) and the Node Operating System (NodeOS) [4] as shown in Figure 1. An EE defines a programming interface that allows users to control the active network; a NodeOS defines a set of basic functions to access and manage the resources of the active node. EEs use the abstractions provided by the NodeOS to build a virtual machine made available to users.

The early development of EEs (e.g., ANTS [9], PLAN [5]) occurred prior to standardization and development efforts for a NodeOS, thus the early EEs made use of traditional

---

*This research is supported by DARPA under contract number N66001-97-C-8512.

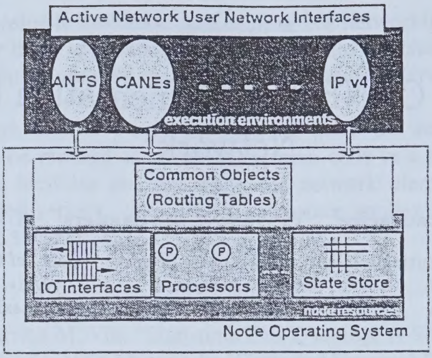


Figure 1: Components of an active network node.

machine operating systems for resource access and management. Experience in developing EEs on top of NodeOS's is, of course, critical for refining the standards being developed. NodeOS projects (e.g., Janos [6], Joust [7]) are developing implementations that complement our experience. In particular, these projects have largely focused on supporting Java and ANTS, while leveraging existing efforts in high-performance operating systems.

We are building an active network comprised of the CANEs execution environment and the Bowman NodeOS. Bowman is constructed by layering active-network-specific operating system functionality on top of a standard host operating system. The host operating system provides low level mechanisms; Bowman provides a channel communication abstraction, an a-flow computation abstraction and a state-store memory abstraction, along with an extension mechanism to enrich the functionality. The CANEs EE provides a composition framework for active services based on customizing a generic underlying program by injecting code to run in specific points called slots. This paper reports on our experience in implementing Bowman (Section 2, instantiating CANEs on top of Bowman (Section 3), and developing applications within CANEs (Section 4). We conclude in Section 5 with a discussion of lessons learned and open problems.

2 Bowman

Bowman provides three basic abstractions to support active network functionality: *channels*, *a-flows* and *state-store*. In addition to these three abstractions, Bowman has an *extension* mechanism that is similar to loadable modules in traditional operating systems. Using the extension mechanism, additional components of a Bowman node can be loaded dynamically at run-time. This section discusses the basic abstractions, related extensions, packet processing path and Bowman performance.

Channels Channels represent the communication resource exported by Bowman. They are communication end-points that support sending and receiving packets via an extensible set of protocols. Bowman exports a set of functions enabling users (i.e. EEs) to create, destroy, query and communicate over channels that implement traditional protocols (TCP, UDP, IP, etc.) in various configurations. In addition, channels in Bowman can include other forms of processing such as compression, forward error correction on the packets.

For packets that do not require per-flow processing, Bowman provides *cut-through channels*, i.e., paths through the Bowman packet processing cycle that do not incur the overhead of multi-threaded processing.

Globally, Bowman implements a network-wide architecture by providing system support for multiple simultaneous *abstract topologies*, i.e., overlay network abstractions that can be used to implement virtual networks. Abstract topologies are built from *links*, which are an association of two of more channels.

A-flows The a-flow is a primary abstraction for computation in Bowman. A-flows encapsulate processing contexts and user state. Each a-flow consists of at least one thread and executes on behalf of an identified principal. Bowman provides a set of functions to create, destroy and run a-flows.

Efficient packet classification to identify flows is an essential part of flow-specific processing. The complexity of packet classification algorithms depend on (1) the type of matching required (first match, longest prefix match, or *best* match based on attributes of matched fields), and (2) the number of fields and types of headers being matched.

Since the identification of a packet is solely based on the pattern of bits contained (possibly, at any location) in the packet, we have implemented a packet classifier that can match on (1) an arbitray number of fields, and (2) each field at an arbitrary offset in the packet. The classifier can be configured (as a matter of node-wide policy) to operate in one of the three modes: return the first match, return all matches, and return the best match(es) according to a cost associated with each field in the classification rule-base.

The Bowman packet classification algorithm does not depend on the protocols used to frame the contents of the packet. But in order to specify a meaningful (readable) pattern of bits to be used to match incoming packets, we can first teach the classifier about a protocol (e.g., Ethernet) and its keywords (e.g., `src`, `dst`, `proto`) using a small language. We can then use these keywords to encode a pattern that will be used to match the packets.

Bowman also provides a fast timeout routine that supports multiple timers per-a-flow. Using the Bowman timeout routine, we have implemented a Bowman extension for threaded per-a-flow timers. Each a-flow contains one timer thread that executes call-back functions when specific timers expire. The resolution of the timers and the number of outstanding timers are configurable parameters.

State-store The state-store provides a mechanism for a-flows to store and retrieve state that is indexed by a unique key. The Bowman state-store interface provides functions for creating, storing and retrieving data from named state-stores. Using the underlying state store mechanism, Bowman provides an interface for creating named *registries*; such registries provide a mechanism for sharing data between a-flows without sharing program variables.

Packet processing path Figure 2 shows a schematic of the Bowman packet processing path. The figure shows the demarcation between Bowman system code and user code that runs within particular a-flows. Within the underlying Bowman system, there are (at least) three processing threads active at all times: a packet input processing thread, a packet output processing thread and a system timer thread. Depending on the input

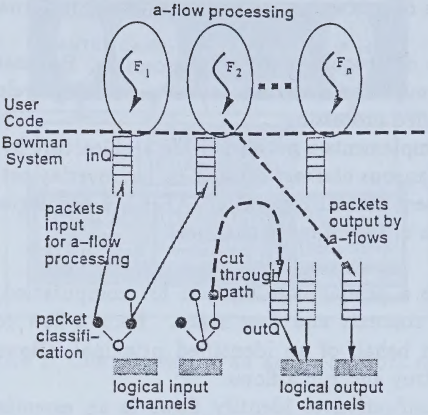


Figure 2: Bowman packet processing path.

and output scheduling algorithms, it is possible for multiple input and output threads to be active within the Bowman.

Packets received on a physical interface undergo a classification step that identifies a set of channels on which the received packet should be processed. In this manner, incoming packets are demultiplexed to specific channel(s) where they undergo channel-specific processing. Once channel input processing completes, the packet is considered to have "arrived" on the abstract link associated with the channel. At this stage, the packet undergoes a second classification step that identifies the further processing (a-flow or cut-through) needed by the packet.

A-flow processing is determined by the code specified when the a-flow is created. Using the Bowman extension mechanism, a-flow code can be dynamically introduced into a node; this mechanism is also used for dynamic code loading (as is required for some active network execution environments).

Bowman performance Our goal in developing Bowman is to leverage the facilities available in traditional operating systems and concentrate on the active-network-specific aspects of a NodeOS. However, we also want to maintain reasonable performance. Bowman has several performance-enhancement features. First, the Bowman implementation is multi-processor capable; on multi-processor machines, Bowman threads may execute concurrently and some thread context-switch times are eliminated. Second, on the Solaris platform, Bowman can be configured to run in real-time mode using the Solaris Real-Time Extensions.[1] With real time extensions, Bowman can deliver high throughput and low delay for packets traversing through a-flows. To leverage the performance advantages of a multi-processor multi-threaded implementation, we have implemented the a-flow input and output queues so that, for single-reader single writer queues, there is no locking (or blocking) unless the queue is empty.

---

[1] See the `priocntl(2)` system call on SunOS 5.5+. An effort is underway to port Bowman real-time extensions to RT-Linux.

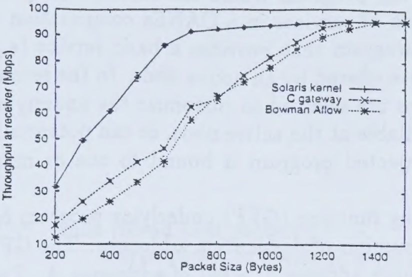


Figure 3: Sustained throughput

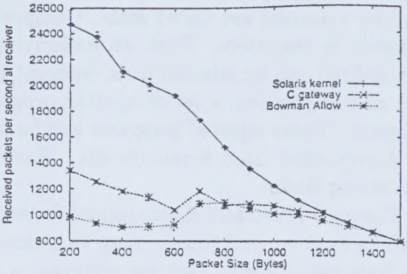


Figure 4: Sustained packet processing rate

We have evaluated the performance of Bowman in a variety of configurations. We include an example result of base forwarding performance here; more detail can be found in [8]. These experiments were performed on a three-node testbed configured in a straight line topology. The interior node executed Bowman. The edge nodes were directly connected to separate 100 Mbps interfaces to the Bowman node. We compare the performance of the Bowman packet processing path with that of the forwarding performance of the Solaris kernel and a gateway program written in C. The single threaded C gateway program uses `socket` calls to read and immediately write out UDP segments. In each case, the sending node transmits UDP segments of a specified size to the destination node. The routing table of one of the edge nodes was configured to use the Bowman node as its next hop to the other edge node in our testbed. The Bowman topology was configured with two bidirectional UDP channels — from the edge nodes to the interior node. (For the C gateway and Bowman forwarding experiments, the Solaris kernel forwarding was turned off).

Figure 3 shows the sustained lossless throughput through Bowman, the C gateway, and the Solaris kernel as measured at the receiver. Figure 4 shows the raw packet rate (in packets/second) at the receiver for the same experiment. It is clear from the figures that the system call, context switch and data copy overheads for small packet sizes precludes both the C gateway and Bowman from sustaining high throughput. However, as packet sizes increase, the context switch and system call dispatch overheads are amortized as more data traverses the system and user space boundary for each system call. Bowman is able to saturate the 100 Mbps ethernet for packet sizes of 1400 bytes or more; Bowman performance is within 5% of kernel forwarding for packet sizes greater than 1200 bytes.

3 CANEs

We have used the Bowman NodeOS as the platform for implementing the CANEs execution environment. This section gives an overview of CANEs, then describes key aspects of the implementation.

Overview The slot processing model used in CANEs EE comprises two parts: a fixed part (*underlying program*) that represents the uniform processing applied on every packet, and a variable part (*injected program*) that represents user-specific functionality on the packets. The specific points in the underlying program where the injected program may be executed are called *slots.* Composition of services in a CANEs computation is achieved in two steps. First, an underlying program that provides a basic service (e.g. forwarding) can be selected from amongst those offered by an active node. In the second step of composition, a set of injected programs are selected to customize the underlying program. These injected programs can be available at the active node, or can potentially be downloaded from a remote site. Each injected program is bound to one or more processing slots.

Figure 5 illustrates a generalized forwarding function (GFF) underlying program for CANEs that supports forwarding to a small number of destination addresses. The GFF has four parameters. Two are required: a source address S, a list of addresses A. Two are optional: a forwarding table identifier R and a selection function M used to match addresses with forwarding table entries. If the optional parameters are not supplied, defaults will be used. The GFF also exports slots where the user may bind injected programs. Each slot has a default behavior, indicated in the square brackets and used if the user does not supply an alternative. For example, Slot 2 is reached if the list of output interfaces turns out to be null. By default, no error message is generated; the user may choose, for example, to send an error message to the source. The other slots provide opportunities to control action taken upon packet receipt and further processing (e.g., sending notification, incoming and outgoing topology information to the source, application of per-interface policy, flow-specific congestion control etc.).

Implementation Figure 6 shows a thread-level abstraction of our CANEs implementation over Bowman. Upon startup, the CANEs EE spawns the control a-flow that contains a thread for handling signaling messages. Each user's computation is executed in its own a-flow and can spawn multiple independent threads that share an address space. The CANEs EE itself provides a library that implements the CANEs API and resides in the system as a single EE a-flow that contains a set of threads to handle housekeeping chores (like signaling and timers).

The CANEs signaling messages are written in the CANEs User Interface (CUI) language and are sent to a well-known port. This message consists of two parts: computation and communication. The computation part defines a directed acyclic graph (DAG). The root of the DAG is an underlying program. Each child node corresponds to an injected program bound to a slot in its parent. The arcs in the DAG identify the slot to which the child is bound. Each node contains enough information for the CANEs EE to fetch the code required to execute the computation DAG. The communication part of the CUI message identifies a routing schema for the user. Further, the user can specify a set of packet classifier rules that are used to select incoming packets that the flow acts on. This communication part provides enough information for the CANEs EE to establish the input and output channels required for the user's computation.

When the underlying program is dynamically loaded, the default entry function "_entry" is executed. This function publicizes the slots exported by the underlying program. Binding of injected programs to the slots is done while parsing the computation part of the CUI signaling message. Invoking a slot is analogous to raising a named event. In this metaphor, all injected programs bound to a slot are handlers for the named event and are executed concurrently when the event is raised.

The CANEs EE provides a variable sharing mechanism that is used by injected programs to communicate with underlying programs. Underlying programs declare shared

```
while (Read next packet) {
    Parse packet to obtain $S, A, [R, M]$
    ⟨Slot 0:[null]⟩
                ; Packet Arrival Slot; Possible uses: route trace, caching
    Outputlist := $\perp$
                ; Initialize the outgoing set of interfaces to null
    For each address $a$ in $A$: {
        Let interface $i$ := Lookup($a, R, M$)
                ; Look up the interface for each address a
        ⟨Slot 1:[Add $i$ to Outputlist]⟩
                ; Code bound to this slot may implement policy
                ; to reject certain interfaces
        }
    if Outputlist = ()
        then ⟨Slot 2:[null]⟩; abort;
                ; When no output interfaces were found, the code
                ; bound to slot 2 can send errors to source
    For each unique $i$ in Outputlist:{
                ; Packet will be dispatched on each outgoing interface
        Create a copy $D$ of the packet,
                with $A' = \{a : (i, a) \in \text{Outputlist}\}$
                ; The new destination address set contains only addresses
                ; that can be reached via interface i
        if $i$ is congested
            then ⟨Slot 3:[discard]⟩
                ; Congestion control algorithms may be bound to slot 3...
        else ⟨Slot 4:[null]⟩
                ; ...alternate schedule policies to slot 4
            enqueue $D$ for $i$.
                ; Eventually, the packet is transmitted on interface i
        }
}
```

Figure 5: Example CANEs underlying program.

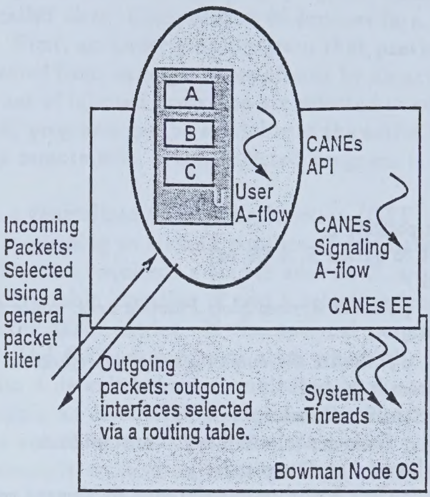

A
B
C
U
CANEs
API
User
A-flow
Incoming
Packets:
Selected
using a
general
packet
filter
CANEs
Signaling
A-flow
CANEs EE
Outgoing
packets: outgoing
interfaces selected
via a routing table.
System
Threads
Bowman Node OS

Figure 6: A high level view of processing at an active node running the CANEs EE over the Bowman Node OS. User flows consist of an underlying program (U in this figure) and a set of injected programs (A, B, and C). This figure also shows a nominal thread level structure of our implementation.

variables and *export* them (make them eligible for sharing). Injected programs declare *references* to shared variables (the space for the actual data for references must be declared by the underlying program). Further, injected programs *import* shared variables exported by underlying programs (this finally creates the binding between the shared and the references variables).

4 Applications

We are in the process of accumulating experience with implementation of applications in CANEs. In this section we describe two moderately complex applications.

Active error recovery

Active Error Recovery(AER) [1] is a joint effort between TASC and the University of Massachusetts to develop a reliable multicast framework using active processing. We have worked with researchers from TASC and University of Massachusetts on an implementation of AER in CANEs.

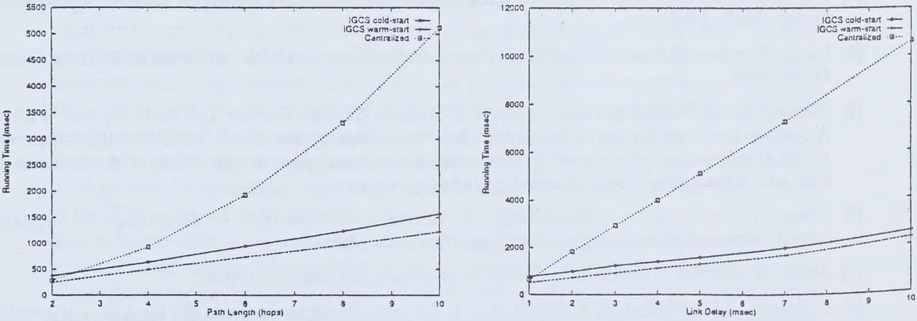
AER makes use of a repair server, residing within the network, to cache packets, respond to retransmission requests, suppress redundant NAKs from receivers, and detect gaps in sequence numbers (indicating lost packets). AER also includes protocols to calculate round trip times and dynamically select a worst receiver to handle sliding-window-based flow control.

We have implemented AER in CANEs using two underlying programs, a multicast forwarding function for sending data, NAKs and source path messages, and the generalized forwarding function for calculating the round trip time and monitoring the congestion status. We make considerable use of the system timers to retransmit and suppress NAKs, to uninstall stale state associated with a flow, and to calculate round trip times.

Iterative gather-compute-scatter

The Iterative Gather-Compute-Scatter (IGCS) distributed computation model provides a mechanism to query and synthesize network state [10]. Such a mechanism is potentially useful for a range of applications that are sensitive to network topology (e.g., placement of an AER repair server). IGCS programs repeat a gather, compute, scatter cycle until a given condition is satisfied. During an iteration, a set of messages are collected during the gather phase. Once a specific set of messages have been collected, the compute phase commences. The inputs to the computation are the set of collected messages, the node and link attributes, and the state store for this computation. The compute phase can produce a single message (of a fixed IGCS message type) that is then "scattered", i.e. transmitted to a set of destinations. IGCS programs can retain state at a node while they are active. All state is lost after the last iteration of the IGCS computation.

The implementation of IGCS consists of an IGCS daemon, an IGCS underlying program and a set of injected programs. The daemon (implemented as a Bowman extension) receives signaling messages and initiates new computations at network nodes. During the initiation of a computation, the underlying and compute slot programs are loaded onto the local node via the Bowman code loading mechanism. The compute slot programs are bound to the proper compute phases. The data part of the signaling message is stored into the local state-store so that the information can be retrieved by the IGCS computation.



(a) Effect of path length

(b) Effect of link delay

Figure 7: Running time with varying link delay and path length

Figure 7 shows the running time of an IGCS application to find the minimum bandwidth on a path. We compare the IGCS implementation to a centralized scheme that must query each node on the path sequentially. In the plot on the left, we vary the length of the path while keeping link delay constant at 5 msec. In the plot on the right, we fix the path length at 10 hops and vary the link delay. For the IGCS algorithm, we show results for both cold-start (i.e., code must be loaded) and warm-start (i.e., code is resident in the cache). This experiment has two salient points. First, the IGCS scheme has a significant performance advantage over a non-active scheme when the number of hops exceeds two. Second, the overhead of code-loading in this example is modest; the cold-start overhead is about 35 msec, while the warm-start overhead is about 5 msec.

5 Discussion

We briefly comment on lessons learned in implementation and on current areas of work. Bowman is intended to be a "shim" layer over a traditional operating system. However, providing useful active networking functionality requires access to low level resources, thus Bowman cannot be insulated from low level details of the OS. Said another way, the "active" part of an active OS (especially one with decent performance) must be reasonably OS-savvy.

We originally designed CANEs so that all slots were raised due to packet arrivals. However, AER (and likely other protocols) require activity that is driven by timers, rather than packet arrivals. We added the ability to raise slots in timer handlers as a result of experience with AER implementation.

We are currently adding to our implementation. Areas of near-term work include better security and protection mechanisms, more sophisticated output queueing schemes, and scalable protocols for virtual topology instantiation.

References

[1] Active Error Recovery (AER). http://www.tascnets.com/panama/AER.

[2] Samrat Bhattacharjee. *Active Networking: Architecture, Composition, and Applications.* PhD thesis, Georgia Institute of Technology, Aug. 1999.

[3] Kenneth L. Calvert (Editor). Architectural Framework for Active Networks. DARPA AN Working Group Draft, 1998.

[4] Larry Peterson (Editor). NodeOS Interface Specification. DARPA AN NodeOS Working Group Draft, 1999.

[5] Michael Hicks, Pankaj Kakkar, Jonathan T. Moore, Carl A. Gunter, and Scott Nettles. PLAN: A packet language for active networks. In *Proceedings of the Third ACM SIGPLAN International Conference on Functional Programming Languages*, pages 86–93. ACM, 1998. Available at `www.cis.upenn.edu/ ~switchware/ papers/ plan.ps`.

[6] Janos: A Java-based Active Network Operating System. http://www.cs.utah.edu/projects/flux/janos/summary.html.

[7] Joust: A Java OS in Scout. http://www.cs.princeton.edu/nsg/joust.html.

[8] S.Merugu, S.Bhattacharjee, E.Zegura, and K.Calvert. Bowman: A Node OS for Active Networks. Submitted to IEEE Infocom 2000.

[9] D. Wetherall, J. Guttag, and D. L. Tennenhouse. ANTS: A toolkit for building and dynamically deploying network protocols. In *IEEE OpenArch'98*, San Francisco, CA, April 1998.

[10] Y.Chae, S.Merugu, E.Zegura, and S.Bhattacharjee. Exposing the network: Support for topology-sensitive applications. Submitted to IEEE OpenArch 2000.

Design of a Flexible Open Platform for High Performance Active Networks

Sumi Choi, Dan Decasper, John Dehart, Ralph Keller
John Lockwood, Jonathan Turner and Tilman Wolf
{syc1, dan, jdd, keller, lockwood, jst, wolf}@arl.wustl.edu

Abstract

This paper describes an architecture for a high performance active router. The system is designed as an open research platform, with a range of configuration options and possibilities for extension in both the software and hardware dimensions. The system is built around a scalable switch fabric and includes a general-purpose processor subsystem at each port, enabling flexible packet processing and deployment of flow-specific *active plugins.* Such a research platform is becoming an indispensable tool for effective systems research in networking and distributed systems.

1 Introduction

In the eighties and early nineties, networking research has concentrated largely on how to improve network performance, by designing switches and routers that could handle large numbers of gigabit rate links. This research has led to the development of systems capable of supporting 2.4 Gb/s links and system capacities in the 100 Gb/s range. Systems now being developed feature scalable multistage interconnection networks and custom hardware processing enabling them to support gigabit links and capacities exceeding 1 Tb/s. While there is a continuing need to improve performance, it appears likely that future developments will be driven primarily by continuing improvements in the underlying electronics technology, accompanied by incremental refinements in system architectures.

Advances in electronics technology affect switches and routers in two ways. First, as the IO capacity of individual integrated circuits grows, it becomes possible to process data at higher rates in a single chip, allowing support of higher speed links and/or more economical processing of data from lower speed links. Second, as the logic capacity of integrated circuits grows, it allows more complex processing to be performed within a single chip. Because the logic capacity of integrated circuits is growing considerably faster than their IO bandwidth, there is a real opportunity to perform more sophisticated processing, without compromising overall system performance. These fundamental trends are already evident in the growing sophistication of switches and routers, which now commonly employ complex queueing and packet filtering mechanisms.

A natural next step in this evolution is the incorporation of specialized programmable components within switches and routers to allow more flexible processing than is possible with custom hardware. Emerging products [3] highlight this direction, providing tens of on-chip processors, enabling gigabit packet processing rates. While the memory limitations of the current generation of devices significantly constrains the range of applications to which

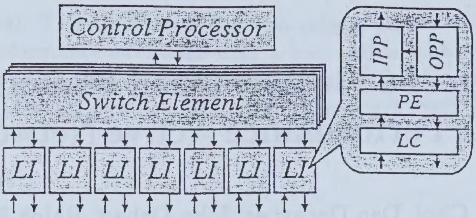


Figure 1: System Organization of Active Router

they can be applied, they clearly show the direction in which the technology is now moving. The introduction of programmable processors with substantial amounts of memory will enable richer processing possibilities, including the embedding of application processing within networked devices. Active networking [13] offers one of the more intriguing visions for how such embedded processing capabilities might be invoked on behalf of users, but it is by no means the only option.

This paper describes an open and extensible research platform for studies in active networking and related areas of networking research. We describe the key hardware and software components making up the system, with an emphasis on the different ways these components can be used. The platform allows new components to be easily introduced, either replacing or augmenting existing components. Such a research platform provides a valuable tool for systems researchers in networking and distributed computing. Because the internal technical specifications are public, the system can be used in innovative ways or modified to enable the provision of new capabilities. Most of the components making up this platform have been distributed to about 30 different universities as part of an NSF-sponsored initiative, and all of the components are now, or soon will be, generally available to the research community [12, 14].

2 System Organization

Figure 1 shows the overall organization of the *Washington University Gigabit Router* (WUGR). At the core of the system is an ATM switch built around a scalable fabric constructed from eight port *Switch Elements* (SE). The base configuration shown in Figure 1 contains a single SE (implemented using four integrated circuits), but the same chips can be used to construct multistage interconnection networks with tens or hundreds of gigabit links [1]. The switch fabric connects to the external links through a set of *Link Interfaces.* Each LI contains an *Input Port Processor* (IPP), an *Output Port Processor* (OPP), a *Processing Element* (PE) and a *Line Card* (LC). The IPPs perform the ATM routing lookup on received cells and the OPPs buffer cells awaiting transmission on the external link. The system supports both virtual path and virtual circuit switching, and both one-to-many and many-to-many multicast. The chip set was originally intended to support link speeds of up to 2.4 Gb/s; however, the current implementation is limited to link speeds of 1 Gb/s. Future versions of the chip set are expected to correct this.

Connections to external links are provided through *Line Cards* (LC) that implement the opto-electronic conversion needed for transmission over fiber optic links and transmission encoding and decoding. Several different line cards have been implemented, including line cards for SONET links at OC-3 and OC-12 rates. The most common line card provides 1 Gb/s transmission rates using an optical data link format similar to Fibre Channel.

Each port on the router has a *Processing Element* card which connects between the

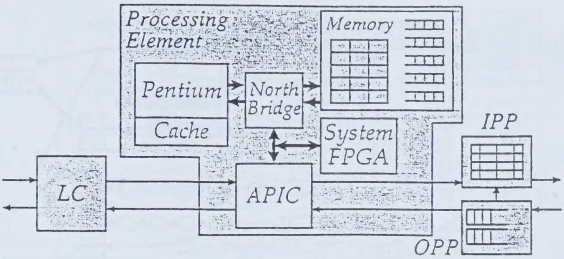


Figure 2: Details of the Processing Element

IPP/OPP and the Line Card, as shown in Figure 2. The PE contains an embedded processor module comprising a Pentium microprocessor (167 MHz), secondary cache, and North Bridge chip, which controls access to memory and which interfaces to the PCI bus used to access peripheral devices. The PE card includes 64 MB of DRAM, a dual port network interface chip for accessing data passing in and out of the system and a *System FPGA*, which implements an interrupt controller, some basic timer mechanisms and miscellaneous other functions that are usually provided by a collection of discrete components in an ordinary PC. The PE card also provides two serial interface ports that can be connected to a computer to assist in debugging the embedded software.

The dual port network interface chip is called the APIC (ATM Port Interconnect Controller). It has two 16 bit Utopia interfaces and a PCI bus interface that can be operated in either 32 bit or 64 bit mode (the PE board uses it in 32 bit mode). The APIC has an on-chip virtual path & circuit translation table and has 256 queues for virtual paths or circuits going to the PCI bus. It pipelines data directly between the network and the processor's memory, avoiding an on-chip packet buffering step, and it supports zero copy transport protocol implementations for maximum performance. The APIC also supports paced transmission of cells from each of up to 256 channels. More details can be found in [8].

The system is managed by a Control Processor (CP), which is implemented using a conventional PC with ample memory and disk. The CP implements routing protocols and other system-wide functions and communicates with the PEs through the switch. It also provides signalling and management interfaces, allowing the system to be embedded in larger testbed networks.

3 Processing Element Software

The Processing Elements (PE) provide both the core packet processing functions and the processing required for active networking. To provide maximal flexibility, the PE software is built around the Net BSD Unix system [9], an open-source Unix system that runs on a variety of platforms, has excellent networking support and a well-structured kernel that is relatively easy to modify and extend.

The kernel software has been augmented using *Router Plugins* [7], a flexible and general tool for extending the functionality of the kernel's packet forwarding path. Router plugins are kernel modules that are statically configured via network management mechanisms and which are applied to particular packets, specified through a kernel-resident packet filter database, which can be dynamically modified. Router plugins have been used to implement *Active Plugins*, which are dynamically configured kernel modules, invoked

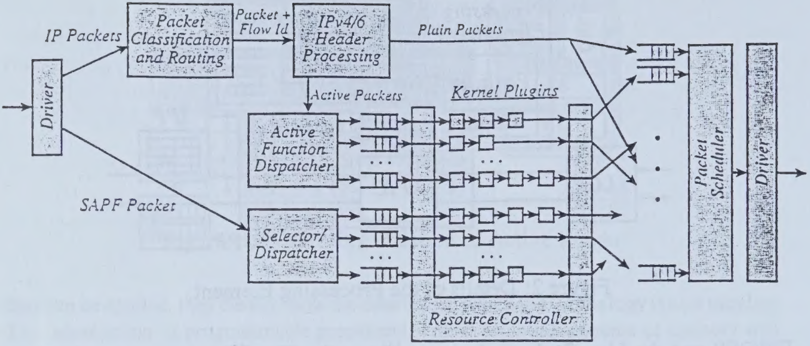


Figure 3: Processing Element Kernel Software

through plugin references carried in active packets moving through the network. The PE software, installs active plugins on-demand, as they are referenced by arriving packets. Plugins that are not available locally are retrieved from one of a set of trusted code-servers. Cryptographic techniques are used to verify that retrieved code does indeed come from a trusted server and to authenticate the identity of the software developer. A policy database is used to control the code servers that are used and/or the developers from which code is accepted. One can also restrict the application of specific plugins to specific packets or classes of packets.

Figure 3 shows the principal components of the PE kernel software involved in processing packets. Packets are assembled in buffers by the APIC and the device driver (the driver prepares the buffer data structures and the APIC transfers and checks the integrity of the data). Packets are then typically passed to a packet classification module that does a lookup based on source and destination address, source and destination port number and protocol. This lookup yields a flow identifier that is used to facilitate flow-specific processing. Packets then pass through an IP processing module that performs standard header processing functions. Non-active IP packets are then placed in one of a set of output queues.

Active packets (identified by a specific protocol value in the IP protocol field) are forwarded to an *Active Function Dispatcher* (AFD), which examines the packet header further to extract the active plugin identifiers that specify what processing that the packet requires. If the required plugins are already present in the kernel, the AFD places the packet in the appropriate queue. The *Resource Controller* (RC) passes packets to the appropriate plugins in accordance with the PE's resource allocation policy, and tracks the amount of processing time used by each packet, in order to regulate future packet selections.

If a requested plugin is not present when a packet arrives, a request is passed to the router's plugin management software. Often, the required plugin will be found in the CP's plugin cache. If not, a request is issued to one or more remote servers to supply the plugin. Upon retrieval, the plugin is configured into the kernel and can be used to process subsequent packets. A packet triggering a plugin retrieval can optionally be forwarded without active processing, queued pending plugin arrival or discarded.

The PE software supports a streamlined processing path for active packets. Packets using the *Simple Active Packet Format* (SAPF) contain a *selector* field which is filled in by the upstream router and used to to enable rapid access to flow-specific state, without the need for the relatively time-consuming packet classification step. The SAPF format also eliminates the need to parse the standard active packet header, which involves extensive

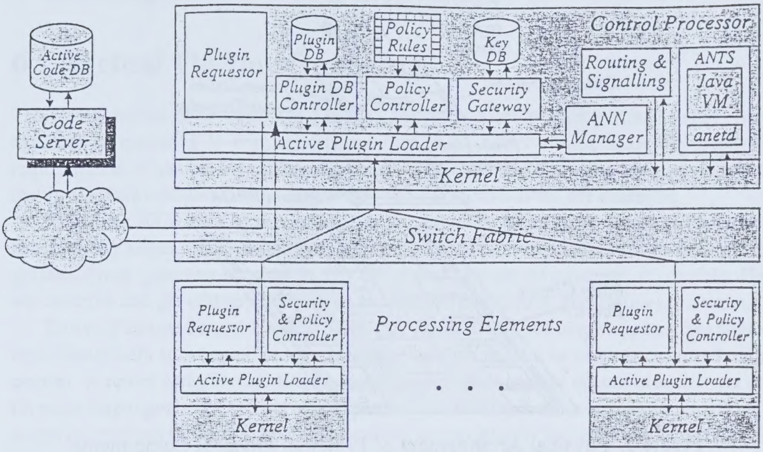

Active Code DB
Code Server
Plugin Requestor
Plugin DB
Policy Rules
Key DB
Control Processor
Plugin DB Controller
Policy Controller
Security Gateway
Routing & Signalling
ANTS
Java VM
anetd
ANN Manager
Active Plugin Loader
Kernel
Switch Fabric
Plugin Requestor
Security & Policy Controller
Processing Elements
Active Plugin Loader
Kernel
Plugin Requestor
Security & Policy Controller
Active Plugin Loader
Kernel

Figure 4: System Level Software Architecture

option processing. The use of the SAPF format is triggered dynamically, using a process similar to *tag switching* [2].

4 System Level Software Architecture

Figure 4 shows the main elements of the overall system software architecture, emphasizing the components involved in plugin management. At the bottom of the figure are PEs, which reside at the switch ports. The Control Processor software is shown at the top of the figure.

When a packet processed by a PE references a plugin that is not already present in the PE's kernel, a request is made to the PE's *Plugin Requestor*, and then forwarded through the switch to the CP, where it is delivered to the CP's Plugin Requestor. The CP's Plugin Requestor checks to see if the plugin is already available in the local Plugin Database, which caches plugins that have been recently used. If the plugin is available locally, it is returned to the requesting PE which loads the plugin into the kernel and continues processing. If the requested plugin is not available locally, a request is sent to a remote code server which responds with the requested plugin, if it is available. Code servers can be arranged into a multi-level hierarchy, and requests that can't be satisfied by one code server are passed up the hierarchy. This allows plugins to migrate from the top of the hierarchy down to the routers that need them.

When the CP receives a plugin, it authenticates the identity of the code server that sent the plugin and the software developer that created the plugin. The policy database allows the router's administrator to specify what code servers and software developers can be "trusted." It also allows limitations to be placed on how plugins can be used (for example, the use of a plugin might be limited to traffic between two specified subnetworks).

As shown in Figure 4, the CP software also includes an *Active Network Node Manager* which enables a router administrator to install a plugin from a remote location. The CP software also includes software to execute network routing and signalling protocols (e.g. RSVP) and software to implement the *Active Node Transfer System* (ANTS) environment, a popular active networking software system that can be used for management and control applications [15].

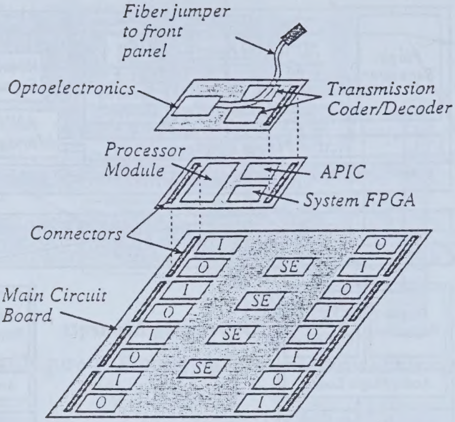


Figure 5: Physical Arrangement of Principal Physical Components

5 Configuration Options

Figure 5 shows the physical arrangement of the different components of the base configuration of the Washington University Gigabit Router. All of these components are packaged within a rack-mountable unit, approximately 12 cm tall, along with a power supply and fans. Links are connected through connectors on the front panel of the unit, which also includes indicator LEDs showing the status of each link.

The core ATM components are contained on a main circuit board that is mounted flat in the bottom of the unit. This board provides eight individual connectors that are used to connect to the PE boards, which are located directly above the main circuit board. These in turn have connectors for the line cards, which are then connected to the front panel of the unit, using fiber optic jumpers. PE cards can be stacked two to a port; or a port can be configured without a PE card, with the Line Card connecting directly to the main board. This is commonly done for the Control Processor port in the base configuration.

The base configuration provides a great deal of flexibility, allowing new components to be introduced at switch ports by creating new cards with the desired functionality and stacking them, as is done with the PEs and the existing line cards. Such cards can use either a 16 bit or 32 bit Utopia interface, although some (including the PE and some line cards) support only the 16 bit interface. While the current line cards support ATM links, it is straightforward to implement line cards for IP-over-SONET or Gigabit Ethernet interfaces as well.

The virtual circuits provided by the ATM core support a variety of configuration possibilities. In the simplest router configuration, there is a single permanent virtual circuit from each input port to each output port. In such a configuration, the PE at an input port forwards the cells of a packet on the appropriate VC. The core ATM components support two priority classes, which can be exploited by a router to provide preferential treatment to some packets (allowing it to support a real-time service with reserved bandwidth, for instance). To exploit this, the system would be configured with two PVCs from each input to each output, one of each priority. Multicast virtual circuits can also be configured, allowing multicast packets to be forwarded to a subset of outputs without the software copying. The ATM layer virtual circuits also provide a convenient mechanism for configuring signaling and network management channels that tunnel through the APICs in the PEs and get forwarded to the CP for processing.

6 Virtual Output Queueing

The bursty nature of IP data traffic makes queueing a central issue in any router. Management of queueing is even more important in an active router, since the bandwidth requirements of packet flows can change as they move through an active network (as a result of active compression or congestion control mechanisms, for example).

While the ATM core of the WUGR router provides some internal cell buffering, the substantially larger memories of the PEs makes them the natural place for queueing in the system. Since queueing is controlled by software, a variety of strategies is possible. Here we describe one general approach as an example of the kind of possibilities that exist.

To avoid congestion within the router, one needs either per output queues that have an input bandwidth substantially larger than the link bandwidth, or one needs *virtual output queues*. A router using virtual output queues, maintains separate queues at each input port for each output port. The rates at which packets are sent from these queues are regulated so as not to overload the output links. To provide approximately the same queueing behavior as a router with true per output buffers, the different input queues sending data to a given output queue should send at a rate which is proportional to the rate at which data enters their queues.

The implementation of per output queueing requires coordination of the sending rates by different inputs to a given output. In small-to-medium sized routers, this can be accomplished by having the PEs at inputs periodically broadcast the rates at which they are receiving data for each output. Using this information, the PE at an output can determine appropriate sending rates for each input and periodically broadcast allowed sending rates for each input. The PEs at an input can then modify their sending rates using the APIC's hardware cell pacing mechanism.

For this approach to work, the control overheads must be acceptably small. If PEs broadcast their requested and allowed rates, say every 100 μs, then each will receive 2 cells every $100/n$ μs where n is the number of ports. For $n \leq 10$, this leads to an acceptably small overhead, if the control messages are processed within the APIC driver. Since a sending period of 100 μs also implies up to 200 μs of delay before a change in input rate is reflected in an increased sending rate, it is important to understand the impact of such a delay. For 1 Gb/s links, we receive at most 25 Kbytes in 200 μs. Since PEs can be configured with several megabytes of buffer space, the delay in adjusting rates is tolerable. Indeed, this example suggests that PEs could use a transmit period as long as a millisecond before the delays in adjusting rates become problematical.

7 Extension for Programmable Hardware

Modern FPGA technology has recently crossed a threshold that enables the rapid prototyping of complex hardware subsystems that once required application-specific integrated circuits. Current devices provide 200 to 300 thousand logic gate equivalents plus 100 to 200 thousand bits of static RAM. It is expected that these complexity levels will double in the next six months. Clock rates of 50-100 MHz can be achieved in complex designs and components with over 500 IO pins are available. Moreover, the design process for FPGAs is substantially simpler and less expensive, than that for ASICs making them ideal for the construction of research prototypes.

These developments open up a rich set of possibilities for hardware extensions to the WUGR router. We are currently developing a *Field Programmable Port Extender* (FPX)

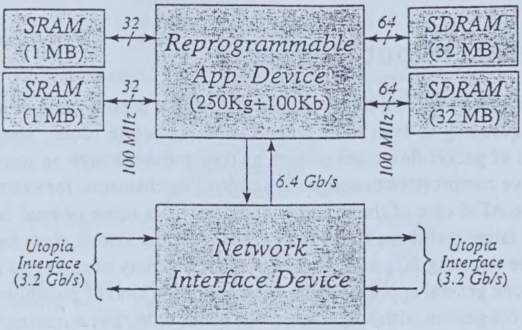


Figure 6: Field Programmable Port Extender

card which will allow new hardware functionality to be added to the system. The FPX will be a stackable card, similar to the PE card. Systems can be configured with one or more FPX cards per port and can mix PE and FPX cards on the same port (subject to constraints on physical space, power and cooling).

The planned configuration of the FPX is shown in Figure 6. The *Network Interface Device* (NID) provides access to the cell stream, much as the APIC does for the PE card. The *Reprogrammable Application Device* (RAD) provides the logic for more complex data processing. It hosts two SRAM and two SDRAM interfaces. The SRAM interfaces are both 32 bits wide and can be operated at a clock rate of 100 MHz. The SDRAM interfaces are 64 bits wide and also operate at 100 MHz. This gives and aggregate memory bandwidth of 2.4 GB/s. This arrangement makes it possible to reprogram the RAD as the system is operating and continuing to forward cells through the NID. While both parts will be reprogrammable, we expect the NID to be relatively static.

The FPX makes it possible to extend the system in a variety of ways. First, IP address lookup and packet classification can be implemented in hardware, relieving the PE software of this time-consuming task. The FPX can also separate packets requiring complex processing by the PE from packets that can be processed entirely by the FPX. This eliminates the bandwidth consumed by such packets on the PE's PCI bus, one of its more constrained resources. The FPX can also implement queueing for ordinary IP packets and can participate in the management of virtual output queues.

The most intriguing possibilities for the FPX involve the implementation of *active hardware plugins*. Such plugins could be used to implement functions that may be too computationally intensive for the PE's general-purpose hardware (encryption, video transcoding). However, implementing this vision requires the solution of a host of thorny issues, including management of the FPGA's hardware resources, coordinating the transition between successive hardware configurations and ensuring that the dynamic installation of one plugin doesn't interfere with the operation of existing plugins.

8 Closing Remarks

The continuing rapid advances in cost/performance of electronics technology is making it possible for network elements to provide ever-more sophisticated processing of data, as it moves through the internet. One of the key challenges for networking and distributed systems researchers is to find effective ways to exploit these new capabilities to enable new network services and applications.

The development of such services and applications requires experimentation in realistic testbed settings, using routers that reflect the architecture and performance characteristics of modern commercial systems. At the same time, academic researchers require flexible, open systems, that can be fully understood and that are easily extended to enable new features. Systems provided by commercial vendors are generally of only limited use to networking researchers, precisely because they are not open for modification by the network research community. This paper has described a flexible, open router that can be an effective tool for systems researchers in search of a more suitable experimental platform.

References

[1] Chaney, T., Fingerhut, A., Flucke, M., Turner, J. [1997]. "Design of a Gigabit ATM Switch," *Proc. of INFOCOM 97*, IEEE, Kobe, Japan.

[2] Cisco Systems, Inc. "Tag Switching: Uniting Routing and Switching for Scalable, High-Performance Services," White Paper, http://www.cisco.com/warp/public/cc/cisco/mkt/ios/tag/tech/tagsw_wp.pdf, 1998.

[3] C-port Corporation. "Communication Processors: a Definition and Comparison," http://www.cportcorp.com/solutions/docs/c5wpaper.pdf, 1999.

[4] Decasper, D., Parulkar, G., Choi, S., DeHart, J., Wolf, T., Plattner, B. "A Scalable, High Performance Active Network Node," *IEEE Network*, January/February 1999.

[5] Decasper, D., Plattner, B. [1998]. "DAN - Distributed Code Caching for Active Networks," *Proc. of INFOCOM 98*, IEEE, San Francisco.

[6] Decasper, D., Dittia, Z., Parulkar, G., Plattner, B., "Router Plugins - A Software Architecture for Next Generation Routers," *Proceedings of ACM SIGCOMM'98*, September 1998.

[7] Decasper, Dan. "A Software Architecture for Next Generation Routers," doctoral dissertation, Swiss Federal Institute of Technology in Zurich (ETHZ), 1999. Available through http://www.arl.wustl.edu/ dan.

[8] Dittia, Z., J. R. Cox, Jr., G. Parulkar. "Design of the APIC: A High Performance ATM Host-Network Interface Chip," *Proc. of INFOCOM 95*, IEEE, Boston, MA.

[9] The NetBSD Foundation. http://www.NetBSD.ORG/, 1999.

[10] Shreedhar, M., Varghese, G. [1995]. "Efficient Fair Queueing using Deficit Round Robin," *Proc. of SIGCOMM 95*, ACM, Cambridge, Mass.

[11] Srinivasan, V., Suri, S., Varghese, G. [1999]. "Packet Classification using Tuple Space Search," *Proc. of SIGCOMM 99*, ACM, Cambridge, Mass.

[12] STS Technologies. http://www.ststech.com.

[13] Tenenhouse, D., Smith, J., Sincoskie, W., Wetherall, D., Minden, G. [1997]. "A Survey of Active Network Research," *IEEE Communications*, 35:1, 80-86.

[14] Washington University Applied Research Laboratory. "Gigabit Networking Technology Distribution Program," http://www.arl.wustl.edu/gigabitkits/kits.html.

[15] Wetherall, David and David Tenenhouse. "ANTS: A Toolkit for Building and Dynamically Deploying Network Protocols," *Proceedings of IEEE OpenArch 98*, April 1998.

[16] Wolf, T., D. Decasper, C. Tschudin. "Tags for High Performance Active Networks," submitted to *OPENARCH 2000*, IEEE, Tel-Aviv, Israel, 2000.

Implementing a Concast Service*

K. Calvert, J. Griffioen, B. Mullins, A. Sehgal, S. Wen
[calvert,griff,mullins,asehgal,suwen]@dcs.uky.edu
Department of Computer Science
University of Kentucky
Lexington, KY 40506-0046

Abstract

Concast is a new network service, the dual of multicast. With concast, messages from different senders addressed to the same receiver are merged by the network into a single message for delivery to that receiver. Like multicast, concast is a scaling mechanism that simplifies application design by allowing a group of peers to be represented by a single (source) address; it is useful in dealing with implosion problems that can arise when many receivers have to provide feedback to a sender. Like multicast, concast conserves network resources by reducing redundant packet transmissions. Unlike multicast, concast is most useful when the application can customize the service by supplying the definition of the "merge" function to be implemented by the network; it is thus an excellent match for active networking. This paper describes a concast framework we have implemented. In spite of its programmability, the implementation is compatible with current Internet protocols, and could be deployed incrementally.

1 Introduction

The Internet is rapidly becoming the virtual workplace where people go to meet, share information, discuss ideas, and, in general, collaborate. Collaborative applications such as teleconferencing, virtual-offices, shared whiteboards, chat rooms, and application sharing are becoming important tools. Information dissemination applications such as stock price updates, network news feeds, mailing lists, and webcasting to interested parties has become the lifeline of many businesses. As a result, flexible group communication models are becoming increasingly important. Although group communication models have been in use for many years [6, 4, 7], past approaches have focused on the problem of one-to-many communication, with extensions to many-to-many. A glaring omission is the lack of network support for many-to-one communication.

We have proposed a new programmable network service called *concast*, which is an inverse or dual of multicast [5]. Multicast uses a single network address to represent a group of

---

*Effort sponsored in part by the Defense Advanced Research Projects Agency (DARPA) and Air Force Research Laboratory, Air Force Materiel Command, USAF, under agreement number F30602-99-1-0514. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of DARPA, the Air Force Research Laboratory, or the U.S. Government.

receivers; concast uses a single address to represent a group of *senders*. When a packet is sent to a multicast address, the network delivers a copy to all receivers who have joined that group. When multiple members of a concast group send packets addressed to a single receiver, only a single packet is delivered to that receiver, and it appears to come from the concast group. Internet multicast is a scalable one-to-many service because the sender need not deal with receivers individually, but instead can treat them as a group. Concast offers similar benefits to applications that require scalable many-to-one communication. Examples of such applications include sensor data collection, group teleconferencing, and any application involving feedback from a multicast group.

A key difference between concast and multicast is the requirement of, and benefit from, having a *programmable* concast service, where applications can *customize the merge semantics* carried out by the network. Applications that can download code into the network to map from the set of sent messages to received message means that any application involving many-to-one communication can potentially benefit from a concast service simply by tailoring the merge function to its needs.

This paper focuses on issues related to the implementation and deployment of a backward-compatible, customizable concast service, based on an early prototype using Linux PCs. Elsewhere we have defined the basic concast service framework and signaling protocol [5]. We begin with an overview of the service abstraction (Section 2) and then describe an implementation in the context of the IP protocol, highlighting problems that must be addressed in this context (Section 3). Section 4 presents the concast API and its implementation. Section 4.3 describes how merge processing can be added to Linux routers to convert them to concast-enabled routers. Finally, we illustrate the utility and customizability of the service with an audio mixer application (section 5).

2 Concast Service Overview

Concast is naturally symmetric with multicast in many ways. Like multicast, concast is a best-effort service, and requires that a portion of the IP address space be set aside for *concast group addresses*. In multicast, the sender does not know who the receivers are; in concast the receiver does not know who the senders are. When a packet is sent to a multicast address, the network delivers a copy to all receivers who have joined that group. The network *duplicates* multicast packets, so a single send results in multiple copies of the sent message being delivered; with concast, the network applies an application-specific, semantics-based *merge* function to map multiple sent messages into a single received message.

The concast receiver is responsible for providing the network with a *merge specification* that defines the mapping from sent messages to the delivered message. The merge specification includes a flow specification that describes the packets belonging to the flow, and descriptions of the various computations that must be carried out on the packets to merge them with other packets belonging to the flow. Each concast sender is responsible for signaling its intent to send as a member of the group (see Section 2.2). Signaling causes the merge specification to be pulled from the receiver toward the sender.

As concast datagrams travel through the network, they are processed hop-by-hop. At each concast-capable node, concast packets (flagged by the Router Alert IP option [9] and a concast address in the source field) are diverted for merge processing. Packets are classified into flows based on source (concast address), destination, and other application-supplied information (Section 3); the relevant merge specification is retrieved; and the flow-specific processing defined by the merge spec is applied.

```
void ProcessMessage(Receiver R, Group G, AppID A, IPdatagram m) {
 FlowStateBlock fsb;
 MsgKey k;
 MergeStateBlock *s;

  fsb = lookUpFlow(R,G,A);
  if (fsb != NULL) {
    t = fsb.getTag(m);
    s = fsb.findMergeState(t);
    s = fsb.merge(s,m,fsb);
    fsb.saveMergeState(s,t);    /* replace previous state value */
    if (fsb.done(s))
      forwardMsg(R,G,fsb.buildMsg(s));
  }
} /* ProcessMessage */
```

Figure 1: Network per-packet processing.

2.1 Concast Processing

Concast-capable nodes maintain a *Flow State Block* (FSB) for each active concast flow. The FSB contains the merge specification, processing state information for any in-process merges, and an *Upstream Neighbor List* (UNL) containing addresses of concast-capable nodes for which this node is the next downstream hop towards the receiver of the flow (R). The hop-by-hop processing of concast messages is defined in terms of the following functions, which together with the flow specification make up the *merge specification*:[1]

getTag(m): a *tag extraction* function returning a hash or key identifying the message. Messages m and m' are eligible for merging iff *getTag*(m) = *getTag*(m').

merge(s, m, f): the function that combines messages together. The first parameter is the current *merge state*, which contains information representing messages that have already been processed. The second parameter m is the incoming message, which will be merged into the saved state s. The third parameter is the flow state block for the concast flow to which m belongs.

done(s): the *forwarding predicate* that checks s, the current merge state, and decides whether a message should be constructed (by calling *buildMsg*) and forwarded to the receiver. This function may set up timeouts to arrange for later processing in case additional packets do not arrive.

buildMsg(s): the *message construction* function, which takes the current merge state, s, and returns the payload to be forwarded toward the receiver.

The processing applied to each concast packet hop-by-hop is shown in Figure 1. When the node detects an incoming concast packet, it first retrieves any state associated with the given flow, which is identified by the receiver (destination IP address), group (concast address), and Application ID (carried in an IP option, see Section 3). If state exists, the identifier (tag) of the message being processed is computed using the receiver-supplied *getTag*() function, and any associated message state is retrieved. A new message state block is computed from the old state and the message payload, and the result is stored back with the tag. If the *done*() predicate indicates that merging is finished, the *buildMsg*() function is called to generate a payload, which is forwarded toward the receiver with the concast source address and the IP address of the receiver for this flow.

---

[1] We do not deal with the issue of a language for specification of these functions by the receiver, except to note that languages for specifying code in active networks are the subject of ongoing research [8, 13, 11].

2.2 Concast Signaling

The *Concast Signaling Protocol* (CSP) deals with the establishment of flow state in the concast-capable nodes of the network. Here we merely outline the basic CSP operation to show its simplicity; see [5] for further details.

CSP messages are sent as regular IP unicast datagrams with CSP identified in the protocol field, and the Router Alert IP option included to stimulate hop-by-hop processing. The signaling process begins when a receiving application indicates to the local concast module its desire to receive from a specific concast group, and supplies a merge specification. The local concast module creates a FSB for the flow with an empty UNL and installs the merge specification. On the other end of the flow, before sending a concast packet from group G to R, a concast sender first signals its intention to do so. This triggers the local concast module to check whether there already exists a FSB for (R,G). If not, the local module creates a FSB for the flow and sends a *Request for Merge Specification* (RMS) signaling message addressed to the receiver.[2] As the RMS message travels toward the receiver, it is intercepted by each concast-capable node along the way, until it reaches a node with the merge spec. Each node along the way creates an FSB, adds the source address to the UNL for the flow, inserts its own address as the source, and forwards the request.

When the message reaches a node that has the merge specification installed — in the worst case R — that node replies with a *Merge Specification* (MS) message containing the spec supplied by the receiver (encoded in a platform-independent way). Thus the merge specification propagates from the receiver to the senders. New senders are "grafted" onto the concast tree at the point where their path to the receiver intersects the existing tree (which may be at the receiver itself). Intermediate nodes that have only a single upstream neighbor do not invoke the merge function on incoming messages, but forward them immediately for efficiency.

Each concast-capable node periodically sends a "KeepAlive" message downstream for each of its active concast flows. If a node fails to receive a "KeepAlive" message from an upstream neighbor node for several periods in a row, it "prunes" that branch of that flow by removing the node from the flow's UNL; thus inactive senders are eventually removed from the tree. When a flow's UNL becomes empty, its FSB is deallocated. Errors are handled by sending CSP *Error* messages upstream. For example, if an RMS message reaches the receiver's node and the receiver has not provided a merge spec for the given flow, an Error message indicating "No Such Merge Spec" is propagated back upstream to the originating sender.

3 Issues

A number of issues arise in the design and implementation of a backward-compatible concast service. In this section we outline them and briefly describe the approach we have chosen.

Identifying Concast Flows. As soon as application-specified semantics are admitted, the possibility arises that different receiving applications on the same host will need different forms of the service and will supply different merge specifications. The network has to be able to distinguish among these flows so that the function `lookUpFlow` (Figure 1) returns the proper flow state block for each incoming datagram. At least two solutions are possible. One identifies a flow with the pair (receiver IP address, concast group); the other identifies it with the four-tuple (receiver IP address, receiver protocol, receiver port number, concast group). The first approach does not require the network to be aware of higher-level protocols, but requires that different applications on the same host use different concast addresses when they receive from the same concast group. The second approach allows the concast address to represent the group of senders, independent of receiver, but violates layering and relies on having port numbers in all transport protocols in the same place.

---

[2] The source address of a CSP packet is always the (unicast) IP address of the packet's origin; this is crucial because it enables connectivity problems in the concast tree to be detected via the Internet Control Message Protocol. (ICMP messages cannot be sent in response to packets with concast source addresses.)

We decided to implement a third approach that borrows from the strengths of both. In our approach each concast service is identified by a tuple (G, R, A), where G is the concast group address, R is the receiver's IP address, and A is an ID that uniquely identifies the receiving application.

G and R are of course carried in the IP header; to avoid depending on application identification being in a particular location in the IP payload we place the application ID A in an IP option. The "AppID" option contains an opaque 32-bit number. How the AppID is chosen (and agreed upon between senders and receivers) is not defined by the concast mechanism; one simple way to ensure uniqueness is to derive the AppID from the receiver's protocol and transport address, which must be known to all parties anyway.

Flow State Scalability. Concast requires that nodes maintain state information for each flow (receiver and group). For scalability it is necessary to limit the per-flow state to a bounded (preferably fixed) size. Enforcement of this requirement is closely tied to the form of the programming interface. If merge specification functions are provided in the form of Java bytecodes, for example, the virtual machine used to execute those functions needs to ensure that their resource usage is limited. (Various approaches to this problem have been proposed in the context of active networks [13, 8].) For example, only a fixed number of in-progress merge states could be kept for any particular flow, with the size of each limited to a network-wide bound. (Note that this implies a limit on the rate at which the network can merge packets for any particular flow.) Given a bound on the amount of per-flow state, concast is approximately as scalable as multicast, which requires routing state for each (source, group) flow.

Fragmentation. So far our description has assumed that senders' IP packets were not fragmented. However, if the sender transmits large IP datagrams, routers along the way may split them into fragments, spreading the original payload across the fragments. In general, fragmentation is a problem because merge functions will typically be designed to operate on full packets.[3] Possible solutions include reassembling fragmented datagrams before handing them off for concast processing (which is likely unacceptable due to processing load, delay, etc); path MTU discovery by the senders; and having the application do its own fragmentation and reassembly, of which the merge function is aware. Our recommended approach is per-sender PMTU discovery.

4 Implementing the Concast Framework

To evaluate the benefits of the concast service model, we are implementing a prototype concast framework using Linux PCs as the development platform. In the following we describe our design and illustrate where concast processing fits in the standard protocol stack. Our discussion is couched in terms of the Unix operating system, but a similar approach could be followed with most any operating system (particularly ones that support sockets). Our current prototype does not yet implement the CSP signalling protocol, and thus, merge functions must be installed manually at routers rather than being pulled down automatically. However, the signalling protocol described in section 2.2 can be implemented in our framework in a straightforward manner.

---

[3] Designing the *getTag* and *merge* functions to operate on fragmented datagrams requires that the merge operation commute with reassembly, that is, the result of reassembling the merged fragments is the same as the result of merging the reassembled fragments at each hop.

4.1 The Concast Programming Interface

Applications on the end-systems enable concast service by creating sockets in the normal fashion, and then associating the socket with a concast flow. Two new system calls are provided to associate a socket with a concast flow: `CCAST_JOIN_FLOW` and `CCAST_SET_MERGE`. `CCAST_JOIN_FLOW` is invoked by senders, while `CCAST_SET_MERGE` is invoked by receivers. Both calls are implemented as `setsockopt()` calls as shown in Figure 2.

```
setsockopt(sock, IPPROTO_IP, CCAST_JOIN_FLOW, &flowspec, sizeof(flowspec) )
setsockopt(sock, IPPROTO_IP, CCAST_SET_MERGE, &mergespec, sizeof(mergespec) )

    struct flowspec {                  struct mergespec {
      ulong ConcastAddress;              struct flowspec fs;
      ulong ReceiverIPAddress;           char *getTag;
      ulong AppID;                       char *merge;
      ushort ReceiverPort;               char *done;
    }                                    char *buildMsg;
                                       }
```

Figure 2: The `CCAST_JOIN_FLOW` and `CCAST_SET_MERGE` system calls.

The `CCAST_JOIN_FLOW` call takes a flow specification and sets up the socket for concast communication. Specifically, it binds the socket to the specified source and destination addresses. All packets sent via this socket will have `ConcastAddress` as the source address and `ReceiverIPaddress` as the destination address; if the protocol supports port numbers, `ReceiverPort` will be used as the destination port number. Because the endpoints of the socket are fully specified, calls such as `sendto()` and `receivefrom()` are not permitted. In addition, the socket is configured so that the IP Router Alert and the IP AppID options will be placed in the IP header of each outgoing packet. The method of computing the AppID value (see section 3) is up to the application; in our audio application (see section 5) we set $AppID = (Protocol << 16) \mid ReceiverPort$. `CCAST_JOIN_FLOW` also initiates the CSP signalling protocol as described in Section 2.2. The call blocks until a Merge Specification or Error message (e.g. No Such Merge Spec, Destination Unreachable), is received for the flow.

The `CCAST_SET_MERGE` call binds the socket to a flowspec, but also installs the merge functions needed at concast-enabled routers. Specifically, it registers the socket to receive packets belonging to the flow (i.e., packets with `ConcastAddress` as source, destination IP address `ReceiverIPaddress`, and IP Application Id option with containing value `AppID`. (It also binds the receiver to the local port `ReceiverPort`.) It also causes the specifications for `gettag`, `merge`, `done` and `buildmsg` —here we assume they are supplied as character strings— to be parsed and compiled/installed locally so that they can be applied to incoming messages. The specifications themselves are stored with the FSB, for transmission in response to subsequent CSP Request for Merge Specification messages.

4.2 End-System Processing

The sender and receiver protocol stacks must be modified in order to transmit and receive concast packets. Figure 3 illustrates the protocol stack organization used at both the sender and receiver.

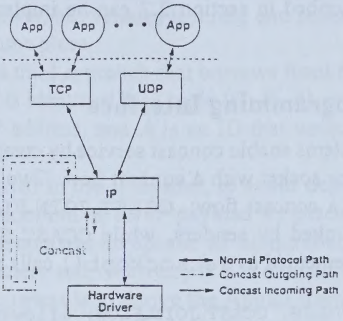


Figure 3: Sender/Receiver protocol stack organization

The concast module at the end system interfaces with the IP module, and is responsible for carrying out merge processing whenever the end system is a merge point for a flow (e.g. two senders on the host sending on the same flow, or incoming branches from different upstream neighbors destined for a local receiver). Flow identification is based on the flowspecs set via the `setsockopt()` calls. Outgoing concast packets are handed off to the concast module during IP processing for insertion of a concast source address, router alert option, AppID option, and merge processing if applicable (i.e. if the flow has other senders on the same host). Incoming concast packets are pulled off the normal IP processing path by the IP router alert option and passed to the concast module for merge processing. The concast module applies the merge functions specified in the `setsockopt()` calls, saving the merge state in a FSB until a merged packet can be given to the IP layer for delivery. When processing is finished, concast packets are then returned to the IP module for forwarding—either transmission on a network interface or delivery to a higher-level protocol. Similar layering organizations have been used by other protocols such as IPSEC [12, 10].

Our current implementation supports the `setsockopt()` calls required to bind the socket to the specified flowspec. Outgoing packets are marked with the IP Router Alert. Merging at the sender is not yet implemented, so only one (G,R,A)-bound socket per host is allowed. The merge function is applied at the receiver, but is currently applied at user-level inside the receiving application.

4.3 Concast Router Implementation

Merge processing at the concast routers (Linux machines) occurs in user-level processes, each executing in its own protected address space. Because we have not implemented the concast signalling protocol, merge functions must be manually started for the concast flow. Merge functions are currently written in C/C++ and linked into the concast daemon, which invokes them via calls to *getTag*(), *merge*(), *done*(), and *buildMsg*(). The daemon maintains a flow state block (FSB) for each active flow, storing the per-tag message state used by the *merge*()

and *buildMsg*() functions. To allow the routers to intercept concast packets, we modified the Linux kernel to direct packets with IP router alert and concast source addresses to a kernel module that demultiplexes the packet based on (G, R, A), where A is obtained from the IP AppID option.[4] Each concast daemon executes a flow-specific merge function. It opens a raw IP socket for receiving packets and issues a `setsockopt()` call that we have added to the Linux kernel which registers the daemon's interest in all concast packets (with Router Alert set) from flow (G, R, A). The *buildMsg*() function forwards packets using a socket that routes packets in the standard way (based on destination address).

5 An Audio Merge Function

A research area that has seen a flurry of activity in recent years is the problem of transcoding network flows [1, 2, 3]. The problem of transcoding arises in a wide range of contexts. However, the basic problem is the same: a flow of data from a sender to a receiver must be modified in some way at intermediate nodes in the network. Consequently, transcoding involves placing application-specific processing in the network to modify the flow of data. When multimedia flows converge in the network upstream of a link that has insufficient bandwidth for the load represented by the individual streams, a typical solution is to place a combiner at the point of convergence. Such a combiner, which takes in individual media streams and emits a single stream carrying the aggregated signal, is a natural application for concast. In the following we look at one such application, an audio combiner that takes in multiple μ-law-encoded (8000 8-bit samples/sec) streams and emits a single such stream, carrying the sum of the incoming signals.

We implemented an audio transcoding system of this type and measured its effectiveness using our concast framework. Our simple audio mixing function converts incoming μ-law streams to 14-bit linear samples, sums them, and converts the result back to a 64 Kbps μ-law stream; peaks that exceed the dynamic range are clipped. In our simulations, senders buffer 1000 samples and transmit them in a single packet every 125 ms. Each outgoing packet contains a sequence number identifying the timing interval to which the packet corresponds. The receiver and intermediate merge points in the network expect to receive a packet from every upstream neighbor about every 125ms. To ensure proper synchronization of the voice streams, each merge point maintains a variable n_i indicating the "next sequence number expected" from each incoming stream i. (Note that incoming streams may already have been merged upstream.) Every 125ms, the n_i's are incremented. When a packet arrives on stream i its sequence number p_i is checked against n_i. If the packet is late, the merged sample for its interval has already been transmitted; in that case $p_i < n_i$ and the packet is discarded. If $p_i > n_i$, the packet is early; it is buffered and merged into the proper interval. Because clocks drift, the combiner automatically adjusts sequence numbers for an upstream neighbor that consistently runs behind or ahead.

We installed the audio mixing system as merge function on a network of Linux boxes that support the concast service. We used the topology shown in Figure 4 to emulate a system with a shared bottleneck link. The audio merge function was placed on the concast routers G_1 and G_2. We created simultaneous audio flows by starting sender applications (processes) at S_1 and S_2, so that two streams were sent from each toward the receiving application at R_1 (a total of four 64 Kbps streams). The concast routers G_1 and G_2 were each initialized with the audio mixer *merge*() function and a *getTag*() function that examined the sequence number. G_1 merges flows from S_1 and S_2 as described above, while G_2 just passes the merged flows through unchanged. For our tests, a unique A value was sufficient to identify each flow and all FSBs were loaded with the audio mixer function described above.

---

[4] Note: the results shown in Section 5 were taken from our first implementation, which performs the (G, R, A) demultiplexing in the concast daemon rather than in-kernel.

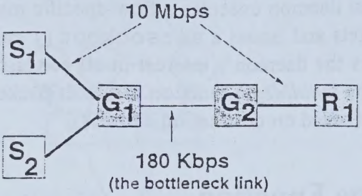


Figure 4: Network topology used in the audio mixing experiments.

To compare a concast-based implementation with the conventional unicast based implementation, we also implemented the audio mixer application using unicast messages sent to the receiver application. In this case, merging only occurred at the receiver.

Figure 5 shows the average number of senders included in the final audio stream delivered to the receiving application over each 1-second interval of the experiment. Results are shown for both the concast and unicast implementation. As expected, merging at G_1 ensured that no flow consumed more than 128 Kbps was sent over the bottleneck link, and thus concast was able to support all four simultaneous flows without losing audio from any of the senders. In contrast, the unicast implementation imposed a load of 256 Kbps on the bottleneck link, resulting in high loss rates. The erratic nature of the unicast delivery made it very difficult to maintain synchronization; occasionally multiple packets from the same source were included in the merged signal. This is the explanation for the spike above 4 at around 55 seconds in the unicast curve. In each experiment, we had two of the senders stop transmitting before the other two; thus both curves drop from 4 to 2 senders toward the end.

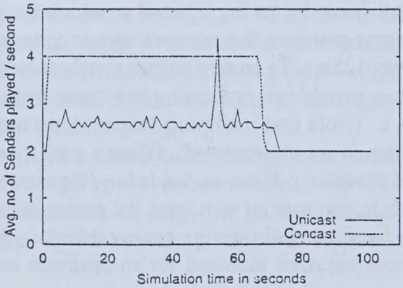


Figure 5: One-second moving average of the number of senders' signals played out at R_1 in each 125 ms interval.

6 Conclusions

Services that can be customized through a programming interface are likely to play a key role in future networks, although the forms of programmability that will be important are yet to be determined. We have defined a programmable network service called concast, which provides scalable many-to-one communication and allows applications to customize its semantics for merging messages. Our previous work demonstrated the benefit of concast through simulation [5]; currently we are implementing concast in the Linux kernel, and we expect to make that implementation available in the near future.

References

[1] The UCL Transcoding Gateway (UTG). http://www-mice.cs.ucl.ac.uk/multimedia/projects/utg/.

[2] E. Amir, S. McCanne, and R. Katz. An Active Service Framework and its application to Realtime Multimedia Transcoding. In *Proceedings of the ACM SIGCOMM '98 Conference*, Sept. 1998.

[3] E. Amir, S. McCanne, and H. Zhang. An Application-Level Video Gateway. In *Proceedings of the ACM Multimedia Conference*, Nov 1995.

[4] Ken Birman, Andre Schiper, and Pat Stephenson. Lightweight Causal and Atomic Group Multicast. *ACM Transactions on Computer Systems*, August 1991.

[5] K. Calvert, J. Griffioen, A. Sehgal, and S. Wen. Concast: Design and implementation of a new network service. In *Proceedings of 1999 International Conference on Network Protocols, Toronto, Ontario*, November 1999.

[6] David R. Cheriton and Stephen E. Deering. Host Groups: A Multicast Extension for Datagram Internetworks. In *Proceedings of the 9th Data Communications Symposium*, pages 172–179. ACM/IEEE, September 1985.

[7] David. R. Cheriton and W. Zwaenepoel. Distributed process groups in the V kernel. *ACM Transactions on Computer Systems*, 3(2):77–107, May 1985.

[8] Michael Hicks, Pankaj Kakkar, T. Moore, Carl A. Gunter, and Scott Nettles. PLAN: A Packet Language for Active Networks, 1998.

[9] D. Katz. IP Router Alert Option, February 1997. RFC 2113.

[10] S. Kent and R. Atkinson. Security architecture for the internet protocol, November 1998. Internet Request for Comments 2401.

[11] B. Schwartz, A. Jackson, W. Strayer, W. Zhou, R. Rockwell, and C. Partridge. Smart Packets for Active Networks. In *1999 IEEE Second Conference on Open Architectures and Network Programming*, pages 90–97, March 1999.

[12] D.A. Wagner and S.M. Bellovin. A Bump in the Stack Encryptor for MS-DOS Systems. In *The Proceedins of the 1996 Symposium on Network and Distributed Systems Security (SNDSS'96)*, 1996. http://bilbo.isu.edu/sndss/sndss96.html.

[13] D. Wetherall, J. Guttag, and D. L. Tennenhouse. ANTS: A toolkit for building and dynamically deploying network protocols. In *IEEE OPENARCH'98*, San Francisco, CA, April 1998.

Active Distributed Management for IP Networks

Ryutaro Kawamura
NTT Network Innovation Laboratories
Currently: Visiting Researcher
Center for Telecommunications Research
Columbia University, New York
ryutaro@exa.onlab.ntt.co.jp

Rolf Stadler
Center for Telecommunications Research and
Department for Electrical Engineering
Columbia University, New York
stadler@ctr.columbia.edu

Abstract

We argue that a management platform for the future Internet has to be inherently distributed and programmable. This motivates us to introduce a new management architecture, named *Active Distributed Management (ADM) architecture*, which exploits the Active Network and Mobile Agent paradigms and provides the properties of distributed execution and programmability. We realize the ADM architecture as a management middleware, composed of several layers. In order to facilitate the development of efficient and correct programs, these layers include patterns for distributed algorithms that are typical for management applications and a set of building blocks for constructing management programs. First results of an ADM prototype system are given and a number of IP management applications are presented.

Keywords: Network Management, Active Networks, Mobile Agents, Distributed Management, IP Network Management

1. Introduction

The scale of the worldwide IP network, the Internet, is increasing at a great rate and is about to take over the role of the primary communication infrastructure from the conventional Telecom (POTS) network. The current Internet service model is simple, a "best-effort" datagram service. In such an environment, the need for network management functionality is limited, basically restricted to element management and some diagnostic end-to-end capabilities. However, current developments are changing this situation dramatically: the future Internet is envisioned as a multi-service platform, supporting real-time traffic and complex multiuser services. It is generally seen as the communication backbone of the emerging digital society. On the technical level, these developments imply that the current Internet model needs to be extended, to include QoS guarantee and differentiation mechanisms (e.g., Intserv, Diffserv), resource reservation schemes (e.g., RSVP), reliability features, a security framework, virtual private networking services, and an SLA (service level agreement) architecture enabling interactions among ISPs. In contrast to the current Internet, this extended Internet model requires network-wide, end-to-end control and management support.

We believe that traditional telecom solutions are not suited for Internet management. A characteristic of those solutions is that they rely on a centralized, hierarchical scheme for control, where management programs control and coordinate the execution of commands on the network element level, often through several levels of a hierarchy. For effective Internet management, we see the need for an *inherently distributed management architecture*, for the following reasons. First, we envision that the number of network nodes in an Internet administrative domain (such as an ISP domain) will be in the order of 10 times the number of nodes in a telecom domain today (such as a domain providing POTS service). This implies that a centralized scheme for management control operations will hardly be scalable and will not allow for efficient operation. Second, research in the telecom field itself has recognized the drawbacks of centralized management schemes and has developed several approaches to distributing management tasks, starting from the concept of management by delegation to the application of mobile agents (see section 2). While we feel that this research goes into the right direction, we think that further advances towards distribution are needed for effective Internet management.

Currently, two types of developments in the Internet community with respect to distributed management can be observed: first, the activity centering around Extensible Agents, with the goal to distribute agent functionality and to provide transparent access to subagents [19]; second, the work on "functional MIBs," such as expression MIB, Event MIB, Scheduling MIB, Script MIB, and Remote Operations MIB [15]. Both of these developments, while delegating functionality towards network elements, still assume that the control and coordination of a management task ultimately resides within a management program, communicating with one or more agents.

In addition to allowing for distributed operation and control, we think that a management platform for the future Internet needs to be *highly programmable*. We expect that new features and services will be added on a frequent basis, while the Internet develops. This implies continuous addition and extension of management functionality, which can be facilitated by a flexible, programmable management platform. Second, the above mentioned network enhancements, such as end-to-end QoS mechanisms and security functions, will require a host of new management applications, which, again, call for a programmable management platform.

Our approach to developing an Internet management architecture, which is inherently distributed and programmable, is based upon using active technologies. We use the term *active technologies* to include the recently developed paradigms of Active Networks (AN) [1-3], Programmable Networks (PN) [18], and Mobile Agents (MA) [4]. These paradigms all support computational models that utilize distributed computing resources *inside* the network. This way, new types of distributed control applications can be realized. The feasibility of these paradigms has been demonstrated, mostly based on distributed object systems, such as CORBA and Java. Although the original concepts of AN, PN and MA were introduced by different research communities to address different problems, they have started overlapping in focus and applicability, as they are being developed further.

In this paper, we first explain the properties of active technologies, which we find necessary for a distributed and programmable management architecture. Also, we discuss the concept and implication of distributed control, which is a key feature of our approach. Then, our architecture, realized as a middleware platform, is introduced in Section 3. Results from our first prototype are shown in Section 4. Note that the current version of this paper focuses on providing the motivation for our work on a distributed and programmable platform for Internet management, and it details the approach we have worked out. The development of the prototype system has started just recently and, therefore, results presented in Section 4 are limited.

2. Network Management using Active Technologies

2.1 Active Technologies

Different types of active technologies have been investigated over the last several years. For this paper, two of them are of primary interest: Mobile Agents (MAs) and Active Networks (ANs). The Mobile Agent concept is a computational paradigm, characterized by computer programs migrating inside the network and executing on servers or network nodes [4]. These programs, also called agents, are self-controlled, and they decide to move to appropriate locations to execute themselves. The motivation behind this paradigm is to overcome certain shortcomings of server-client computing. MAs are often used when it is beneficial to execute a program near a data source in a distributed environment.

Active Networking is a computational networking paradigm, in which so-called active packets, containing programs and user data, are transported between network nodes, e.g. routers [1]. In the original AN concept, called "capsule approach," an active packet (or capsule) contains a program that is executed on every node along the route of the packet through the network. This way, a flow can be processed inside the network. Applications of this paradigm include packet filtering, congestion control, and service-specific routing. As a result of recent research, the capsule approach has been refined into various flavors of the AN concept [1-3].

While MAs have been applied to the domain of distributed applications, their use in networking has generally been restricted to the management plane. On the other side, ANs have been developed to provide for packet processing in the data path, i.e., in the user plane (and sometimes the control plane). Only recently have AN concepts been applied to the management plane [10, 12].

Although the motivations for proposing the MA and AN concepts were quite different, emerging MA and AN technologies are beginning to overlap, specifically in the management domain. The reason for this is obvious: an active packet can be interpreted as a (simple) mobile agent, which executes on every node along its path through the network. Similarly, a mobile agent that moves from one node to another node can be interpreted as an active packet, containing data and a program, which is sent from one node to another node in an active network.

Drawing from the MA and AN concepts, we have identified the following properties we believe to be essential for constructing a distributed, programmable platform for the Internet management. In lack of a better term, we call them the *properties of active technology (AT)*.

(1) **Programmability of network nodes:** An executable program, which we call *active code*, can be installed on or downloaded to every node of the network (by users or network operators). The code transfer can be realized either via the user plane or via a management-specific network.

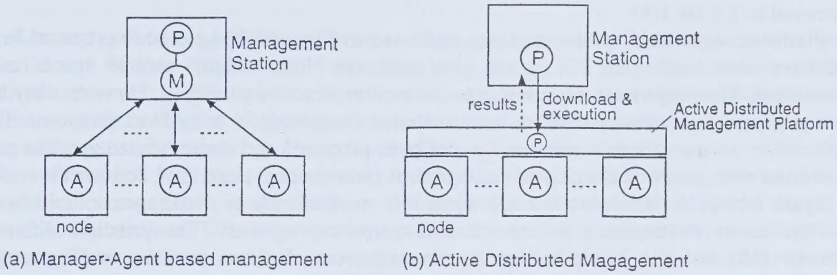
(2) **Autonomous execution of active code:** Active code is executed in a node as an *autonomous, active process*. This implies that an active process does not need to coordinate with a master process, running, e.g., on a management station.

(3) **Mobility and replication of active code:** Active processes can migrate from one node to another node, following autonomous decisions. Moreover, they can spawn child processes. (This property is generally not included in AN architectures, but it is an important characteristic of MA frameworks.)

(4) **Message exchange between active processes:** Active processes can exchange messages inside and between nodes. (Again, this property is generally not included in AN architectures, but it is an important characteristic of MA frameworks.)

Properties (1)-(4) allow us to define a programmable and distributed platform. In the following, we argue that such a platform can be used to run programs that execute management tasks. We propose to realize such a platform as a network-wide management middleware.

2.2 Network management based on active technologies


P
M
Management Station
A
A
A
node
(a) Manager-Agent based management
P
Management Station
results
download & execution
Active Distributed Management Platform
P
A
A
A
node
(b) Active Distributed Magagement

P: Management Program, M: Manager, A: Agent

Fig. 1 Active Distributed Management compared to traditional manager-agent based management

Figure 1 allows us to compare the concept of Active Distributed Management, based on the AT properties (1)-(4), with the traditional style of manager-agent based management. On the left side, Figure 1(a), a management task is executed as a program P on a management station. The execution of P involves manager-agent interactions, whereby P performs operations on the Management Information Base (MIB). The MIB is distributed over a set of agents, which provide access to the network nodes. Note that P exercises *centralized control*, synchronizing the operations of all the agents through its own computation.

The right side, Figure 1(b), shows our approach, which includes an active distributed management platform with the AT properties. In order to run a management program P, this program is downloaded from a management station onto the platform and is started. From that moment on, the management station does not control P anymore. Instead, the control goes over to P itself, which runs as an active process, or a set of active processes, possibly moving around in the network. In other words, the *control of P is inherently distributed.*

While the most important difference between Figure 1 (a) and (b) is centralized vs. distributed control, note the following property of (b), which are not present in (a). Using the active distributed management platform, the program P can be downloaded onto *any* node and its execution will produce the same results. Therefore, the platform we envision processes management programs in a *transparently* distributed way. This allows--to give a simple example--to write a trace-route program, which takes as arguments any two nodes in an IP network.

Most approaches to management, including recent work in distributed management, follow the traditional manager-agent based concept displayed in Figure 1(a). Management by delegation (MBD), for example, allows to download a script from a manager into an agent for remote execution [5]. Note that, in this case, the management station still coordinates the operation. The same argument can be made with respect to other research, such as [14], as well as the activities involving the Script MIB or the Remote Operations MIB [15]. Furthermore, to the best of our knowledge, all applications of the MA paradigm to network management problems use the concept in Figure 1(a). In these instances, a program P typically sends an MA to a network element to perform a task, and the MA later returns with the result. Or, an MA moves from a management station to a network element, installs a piece of software and terminates. Therefore, the use of MAs in the context of management so far has been restricted to one-level hierarchies with centralized control. We acknowledge that the concept in Figure 1(a) allows for distributed execution of a management task. However, the ultimate control of the task is with the management station. This is different from the concept portrayed in Figure 1(b).

Platforms with distributed control, as expressed in Figure 1(b) have been proposed before and have been built. [16, 17]. These platforms use "light-weight" mobile agents, called *messengers*. Messengers are comparable to our notion of active processes. However, they have been proposed as a novel paradigm for distributed computing in a LAN environment. Their applicability to network management has not been proposed and demonstrated yet. The paper of Feridun et al., where a distributed management framework is described, follows the concept in Figure 1(b) [13]. Similar to our approach, this work advocates a management middleware platform as an environment to execute management programs. (The principle difference between [13] and our work is that our management platform provides a programming interface on a much higher level of abstraction.)

We illustrate the main difference between the paradigms in Figure 1 with an example, in which a management program P computes the min, max, and average link utilization in a network. Following the approach in Figure 1(a), P contacts via a manager all agents attached to the network elements, in order to query the utilization of their outgoing links. In a next phase, P processes this data and computes the min, max, and average link utilization. Following the approach in Figure 1(b), P is downloaded to the active distributed management platform and is started. Depending on how P is written, it visits all nodes in the network and continuously computes the data, or it duplicates into several active processes, each of which performs this job on a sub network. The results, then, are aggregated and sent to the management station.

2.3 Discussion of our approach

Having a distributed, programmable management platform with AT properties has advantages when compared to traditional approaches to management, such as those captured by Figure 1(a). Its benefits clearly stem from the capability for distributed control and its high degree of programmability.

(1) Distributed control: Distributed control supported by our platform enables various schemes of distributed processing. It allows, for instance, to process network data near the location at which it is produced--the MBD scheme [5]-- and, in a more general sense, it enables the flexible use of the processing resources inside the network. We believe that, as a result, many management programs can be written in such a way that they scale well with the network size.

Second, management programs written for our platform tend to be more survivable, because there is no central controller (which might fail). Furthermore, the platform is well suited for running self-healing algorithms [20], which, if implemented, provide an additional level of survivability for management applications.

(2) Programmability: The platform can be used to implement almost any type of centralized or distributed algorithm, while, at the same time, giving access to the network elements. Therefore, it is a very flexible environment for developing management programs.

Should the interface to such a platform be published, i.e., made *open*, this could motivate software developers to create innovative and competitive solutions that satisfy the need of network operators and administrators [6].

The biggest challenge we see for realizing a management platform comes from the well-known fact that distributed programs are hard to develop.

Development of management programs: The management platform provides a very powerful and flexible environment for writing applications. On the other hand, developing efficient and correct distributed programs is a difficult task--even more so in a limited amount of time. A program bug or malfunction can have unforeseeable consequences, because it can affect many parts of the network due to the distributed execution of a program.

In order to deal with this problem, our platform has two distinct features, which are being discussed in more detail in the Sections 3 and 4. First, it consists of several layers of functional abstractions. One of these layers contains a library of patterns for distributed programs. These patterns comprise navigation algorithms and policies for parallel execution that are typical for management applications. Examples of such patterns are "moving along a flow in the network" and "visiting all network nodes in a parallel fashion." The idea is that these patterns are being programmed by experts, with focus on efficient and correct execution, and they are given in form of a toolkit to application developers. Second, the platform includes a set of building blocks for constructing management programs, such as collecting node statistics along a certain route. These building blocks are made available to application developers via high-level interfaces.

3. The Active Distributed Management (ADM) Architecture

This section describes the architecture of the active distributed middleware platform. The design objectives of this platform are to facilitate the development of management programs and to provide runtime support for these programs. Although the ADM Architecture is specifically suitable for the Internet and we are developing it for Internet management, it can be applied to telecom networks as well.

3.1 The basic concept

Figure 2 gives a distributed view of the ADM Architecture. Management programs can be launched from a management station that is attached to any of the active nodes. Active code is either installed on the nodes, or it will be downloaded on demand from an active code server. The ADM middleware, shown as a single layer in Figure 2, is displayed as having three sublayers. The functionality of each of these layers is explained in section 3.2.

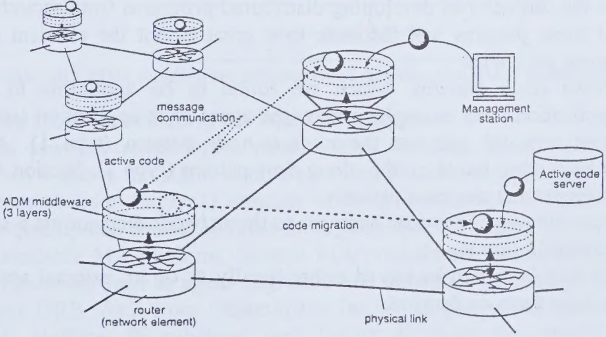


Fig. 2 ADM Architecture overview. (The structure of a single node is shown in Figure 3.)

3.2 ADM middleware

The ADM middleware is divided into three layers, containing functions of different levels of abstraction. Figure 4 shows this model. The details of each layer's functions are described below.

Layer 1: process management layer

This layer realizes the management of active processes, and provides the AT functionality to the higher layers. Table 1 shows representative functions of the process management layer. Since this layer contains the most basic code of the ADMA middleware, it is stored at every node. Most functions of this layer are supported by MA platforms available today. For this reason, we have built our prototype platform using an off-the-shelf MA platform (see Section 4).

Service category	Function
Management of active processes	- create, remove, suspend, resume, duplicate, move
Communication	- communication between active processes (based on Java RMI, CORBA, IIOP, sockets, etc.) - communication type: synchronous, asynchronous, multicast, broadcast
Directory	- location management of active processes
Security	- encryption, authentication, etc.

Table 1 Typical functions in the process management layer

Layer 2: basic operation layer

A management program is executed as one or more active processes, which can move from node to node as part of the program execution. Therefore, we can categorize management programs according to the navigation patterns of their active processes and the number of active processes. (The number of active processes indicates the degree of parallelism of program execution.)

The basic operation layer provides a set of patterns for navigation and different levels of parallelism for management programs. Each of these patterns is realized as an active code fragment. Since the difficulty of developing distributed programs from scratch is well known, we believe that these patterns will facilitate to a great extent the efficient development of robust management programs.

Table 2 shows some patterns, which we found to be applicable to many network management applications. For example, a manager-agent style application (such as collecting information from a node) can use the *node-to-node* pattern (type 1). A route tracing application can be written based on the *along-flow* pattern (type 2). Section 4 contains more examples of programs that use these patterns.

The basic operation layer provides accesses to the network element via a special interface, the NE Management IF (see Fig.3) .

Functions of this layer can be stored either locally or on an external active code server, which allows to load them on demand.

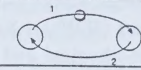
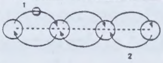
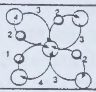
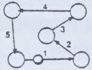
operation	typical application	navigation pattern
type 1: node-to-node	1 node control/monitor (get/set of variables)	
type 2: visit all nodes along a path/flow	1 flow/path control e.g.: traceroute, bottleneck detection, signalling, VPN operation	
type 3: distribute agent to all nodes in subnet (parallel control)	subnet control, message broadcast e.g.: congestion location detection	
type 4: visit all nodes in subnet (sequential control)	subnet control e.g.: topology detection	

Table 2 Typical navigation patterns for management operations

Layer 3: management toolkit layer

The management toolkit layer implements various network management functions, using the navigation patterns provided by layer 2. It gives the highest level of abstraction to the application developers and supports them in writing management programs. This toolkit can be seen as a high-level language for network management. As the code of layer 2, the management toolkit layer can be stored locally or downloaded on-demand from an active code server.

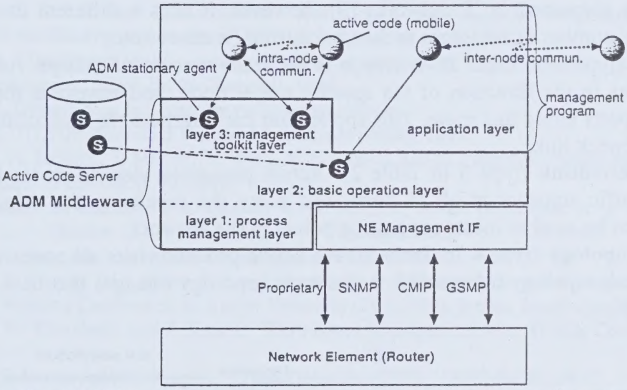

active code (mobile)
intra-node commun.
inter-node commun.
ADM stationary agent
management program
S
S
S
S
S
layer 3: management toolkit layer
application layer
Active Code Server
layer 2: basic operation layer
ADM Middleware
layer 1: process management layer
NE Management IF
Proprietary
SNMP
CMIP
GSMP
Network Element (Router)

Fig.3 Active node with ADM middleware structure and possible interfaces to the network element.

4. Early results from developing an ADM prototype

This section presents some initial results from building an ADM prototype in our laboratory. Figure 4 shows the ADM middleware implementation used in the prototype. The process management layer (layer 1) consists of three sub-layers. In order to develop the prototype more quickly and be able to focus on higher-layer functionality, we decided to use a commercially available MA platform. Several MA products are available, and they support basic MA/AN functions, such as agent migration and object downloading [7]. Our prototype uses the Voyager ORB Ver.3 from ObjectSpace Inc.[8] as part of the process management layer. The MA platform IF sub-layer (see Figure 4) hides the specific MA platform implementation from the higher layers. This means that we will be able to switch the MA platform or implement our own platform in the future, with only minimal changes to the prototype.

The interfaces between layers 1-3 in Figure 4 are implemented using class inheritance. All classes are coded in Java (JDK 1.1.8).

Figure 5 shows the physical and logical network topologies of the current prototype. We installed ADM middleware on eight machines (PCs and SUNs). In the current phase of our development, the ADM nodes emulate the functionality of IP routers. This means that the data we gather from the nodes while running management programs does not come from a real network. However, the current testbed still allows us to investigate and demonstrate the programmability and distributed operation of an ADM platform. In the next phase of our work, we will move the ADM platform onto a real IP network.

We are currently experimenting with the following management programs, which we have implemented on the testbed.

(1) **getinfo** (type 1 in Table 2): **getinfo** is a simple node-to-node type application. An active process visits the target node, gathers specific information, and returns to the original node with the information.

(2) **getpps** (type 1 in Table 2): **getpps** is also one of the node-to-node type applications. It uses the same class as **getinfo** in the ADM middleware layer 2. This application uses active code to measure the level of IP traffic (packets per second) on a specific port for a predetermined time interval.

(3) **traceroute** (type 2 in Table 2): The functionality of this application is basically the same the traceroute command in IP networks [9]; however, it uses a different navigation pattern with a smaller number of messages exchanged during its execution.

(4) **tracepps** (type 2 in Table 2): **tracepps** is the trace version of **getpps**. An active process visits all nodes in the direction of the specific target node, and measures the amount of IP traffic at the ports along this route. This application can estimate the QoS of the flow and can identify bottleneck links.

(5) **findcongestedlink** (type 3 in Table 2): Active processes visit all nodes in the network, gather the traffic statistics at all IF ports, and carry the results to the original node. This application can be used to locate congestion points in the network.

(6) **discovertopology** (type 4 in Table 2): An active process visits all nodes in the network, gathers network topology information, and detects topology changes that have occurred since its last invocation.

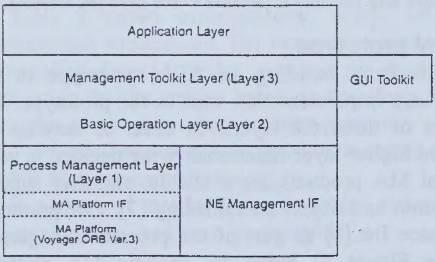
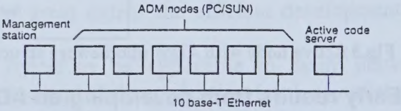
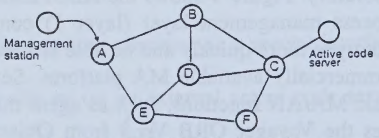

Application Layer
Management Toolkit Layer (Layer 3)
GUI Toolkit
Basic Operation Layer (Layer 2)
Process Management Layer (Layer 1)
MA Platform IF
MA Platform (Voyeger ORB Ver.3)
NE Management IF

Fig. 4 Middleware architecture of the prototype system.


ADM nodes (PC/SUN)
Management station
Active code server
10 base-T Ethernet

(a) physical network topology in laboratory testbed


Management station
A
B
C
D
E
F
Active code server

(b) logical network topology for experimentation

Fig.5 Prototype network topology

5. Conclusion and future work

In this paper, we have proposed a new network management architecture, the Active Distributed Management Architecture. Our architecture utilizes the Active Network and Mobile Agent paradigms, which provide key benefits for a management framework, such as easy distribution of control and flexibility in terms of function updates. In order to deal with the complexity of developing distributed control applications, this paper proposes a middleware solution that includes several layers of functional abstractions. Our approach enables management application programmers to develop customized, robust and efficient management programs within a limited amount of time.

We have developed a first prototype to validate our concepts. However, it only uses PCs connected via Ethernet. We will expand this prototype using actual routers and networks.

References

1. D. L. Tennenhouse, J. M. Smith, W. D. Sincoskie, D. J. Wetherall, and G. J. Minden, "A Survey of Active Network Research" IEEE Communications Magazine, 1997 Jan.
2. K. L. Calvert, S. Bhattacharjee, E. Zegura and J. Sterbenz, "Directions in Active Networks", IEEE Communications Magazine, 1998 Oct.
3. J. M. Smith, K. L. Calvert, S. L. Murphy, H. K. Orman and L. L. Peterson, "Activating Networks: A Progress Report", IEEE COMPUTER Magazine, 1999 Apr.
4. V.A. Pham and A. Karmouch, "Mobile Software Agents: An Overview", IEEE Com. Mag. 1998 July.
5. Y. Yemini, "Network Management by Delegation," Integrated Network Management II, Krishnan and Zimmer, Eds., Elsevier, 1991.
6. A. A. Lazar, K. S. Lim, and F. Marconcini, "Realizing a Foundation for Programmability of ATM Networks with the Binding Architecture," IEEE J-SAC, vol.14, no.7, 1996.
7. T. Magedanz, "Intelligent Mobile Agents in Telecommunication Environments - Basics, Standards, Products, Applications," IEEE IM'99 (Tutorial), 1999.
8. Voyager ORB, ObjectSpace Inc. http://www.objectspace.com .
9. W. R. Stevens, "TCP/IP Illustrated Volume 1," Addison-Wesley, 1994.
10. B. Schwartz, A. Jackson, T. Strayer, W. Zhou, R. Rockwell, and C. Partridge, "Smart Packets for Active Networks," IEEE OPENARCH'99, 1999.
11. D. S. Alexander, B. Braden, C. A. Gunter, A. W. Jackson, A. D. Keromytis, A. J. Minden, and D. Wetherall, "Active Network Encapsulation Protocol (ANEP)," http://www.cis.upenn.edu/~switchware/ANEP/doc/ANEP.txt, 1997.
12. D. Raz and Y. Shavitt, "An Active Network Approach to Efficient Network Management," The First International Working Conference on Active Networks (IWAN'99), Berlin, June/July 1999.
13. M. Feridun, W. Kasteleijn, and J. Krause, "Distributed Management with Mobile Components," IEEE IM'99, 1999.
14. A. Vassilia, G. Pavlou, and G. Knight, "Active Objects in TMN," IEEE IM'97, 1997.
15. J. Schonwalder, "Emerging Internet Management Technologies," IEEE IM'99 (Tutorial), 1999.
16. L.F. Bic, M. Fukuda, M. B. Dillencourt: Distributed Computing using Autonomous Objects, IEEE COMPUTER, August 1996.
17. M. Fukuda, L.F. Bic, M. B. Dillencourt, J.M. Cahill: Messages versus Messengers in Distributed Programming, Journal of Parallel and Distributed Computing, 57, 188-211, 1999
18. A. Lazar, "Programming Telecommunication Networks," IEEE Network, Sept/Oct 1997.
19. M. Daniele, B. Wijnen, D. Francisco: Agent Extensibility (Agent X) Protocol Version 1, RFC 2257, January 98.
20. IEEE Communications Magazine, Special Issue on Survivable Communication Networks, Vol. 37 No. 8, August 1999.
21. R. Kawamura, R. Stadler: Active Distributed Management for IP Networks, CTR Technical Report 469-99-23, Columbia University, 1999.

New Approach for Space-Time Transmitter/Receiver Design*

Hesham El Gamal and A. Roger Hammons, Jr.

Abstract

In this paper, we take a fresh look at the problem of designing multiple antenna wireless systems. First, we cast the problem of space-time signal processing as a multi-user detection problem. We present a set of techniques for joint detection and decoding based on the turbo processing architecture. Then, we present a general architecture that combines algebraic code design with advanced signal processing techniques. This novel technique is referred to as the threaded space-time architecture (TST). The TST provides more flexibility in the trade-off between power efficiency, bandwidth efficiency, and receiver complexity. Simulation results are provided for the various techniques, demonstrating the superiority of the TST compared with both Foshini's layered space-time architecture [1] and Tarokh's multi-layering space-time architecture [2].

1 Introduction

Unlike the Gaussian channel, the wireless channel suffers from multi-path fading. In such fading environments, reliable communication is made possible only through the use of diversity techniques. Recently, information theoretic studies have shown that spatial diversity provided by multiple transmit and/or receive antennas allows for a significant increase in the capacity of wireless communication systems operated in Rayleigh fading environments [3] [4]. Following this discovery, two approaches for exploiting this spatial diversity have been proposed [5] [6] [7] [1]. In the first approach [5] [6] [7], channel coding is performed across the spatial dimension as will as time to benefit from the spatial diversity provided by using multiple transmit antennas. Tarokh *et al.* coined the term "space-time codes" for this coding scheme. One potential drawback of these schemes is that the complexity of the maximum likelihood (ML) decoder is exponential in the number of transmit antennas. The second approach, proposed by Foshini [1], relies upon sophisticated signal processing techniques at the receiver to achieve asymptotically performance close to the outage capacity. In this approach, no effort is made to optimize the channel coding scheme and conventional, *single dimensional* channel codes are used to minimize complexity. This approach is referred to as the layered space-time architecture (LST) [1].

One of the contributions of this paper is a new scheme that combines ideas from these two approaches. Specifically, we propose joint detection and decoding algorithms based on the iterative soft-input-soft-output (SISO) approaches proposed in [8] [9] [10] which avoid the limitations of the LST signal processing techniques, including the need for equal number of transmit and receive antennas. We further present a new *threaded* space-time transmission architecture that benefits from the advantages provided by efficient code design and advanced signal processing. Recently, Tarokh *et al.* proposed a new multi-layering architecture [2] that likewise addresses some of the problems encountered with LST. This approach relies upon a zero forcing group interference suppression technique and shows performance that is 6 – 9 dB from the outage capacity reported in [2]. The threaded architecture and signal processing proposed in this paper close the gap to less than 3 dB from the outage capacity.

2 Signal Model

We consider a multiple antenna communication system with n transmit and m receive antennas. In this system, the channel encoder accepts input from the information source and outputs a coded stream of higher redundancy suitable for error correction processing at the receiver. The encoded output stream is modulated and distributed among the n antennas via a spatial modulator. The transmissions from each of the n transmit antennas are simultaneous and synchronous. The signal received at each antenna is therefore a superposition of the n transmitted signals corrupted by additive white Gaussian noise and multiplicative fading.

The space-time transmission resources may therefore be viewed as an $n \times L$ array whose (i, t)-th entry represents the t-th symbol interval available on the i-th antenna. The dimension indexed by i is referred to as the *spatial* dimension, whereas the dimension indexed by t is called the *temporal* dimension.

The channel encoder is a generic function and, in many cases of interest, will be decomposable into a set of multiple, independent channel encoders processing separate substreams from the information source. When the channel encoder is decomposable, there is a corresponding partitioning of the spatial modulating function that is of interest. In the literature, the components of such a partitioning have been informally referred to as *layers* [1] or *multi-layers* [2].

At the receiver end, the signal r_t^j received by antenna j at time t is given by

$$r_t^j = \sqrt{E_s} \sum_{i=1}^{n} \alpha_t^{(ij)} c_t^i + n_t^j \qquad (1)$$

where $\sqrt{E_s}$ is the energy per transmitted symbol; $\alpha_t^{(ij)}$ is the complex path gain from transmit antenna i to receive antenna j at time t; c_t^i is the symbol transmitted from antenna i at time t; n_t^j is the additive white Gaussian noise sample for receive antenna j at time t. The noise samples are independent samples of zero-mean complex Gaussian random variable with variance $N_0/2$ per dimension. The different path gains $\alpha_t^{(ij)}$ are assumed to be statistically independent. The fading model of primary interest is that of a block flat Rayleigh fading process in which the code word contains B blocks. On each block the complex fading gains are constant but are independent from one block to the next. This model encompasses as a special case the quasi-static fading process studied extensively in [5] [7] in which $B = 1$.

The received signal can be expressed in vector notation as

$$\underline{r}_t = S_t \underline{c}_t + \underline{n}_t \qquad (2)$$

where $\underline{r}_t$ is the $m \times 1$ received vector at time t; S_t is the $m \times n$ complex signature matrix whose i^{th} column corresponds to the path gains for the i^{th} antenna; $\underline{c}_t$ is the $n \times 1$ transmitted vector at time t; $\underline{n}_t$ is the $m \times 1$ white Gaussian noise vector.

3 Generalized Layering

In the layered space-time processing approach, the channel encoder is composite and the multiple, independent coded streams are distributed throughout the transmission resource array in so-called layers. The primary design objective is to design the layering architecture and associated signal processing so that the receiver can efficiently separate the individual layers from one another and can decode each of the layers effectively. In these schemes, there is no spatial interference among symbols transmitted within a layer (unlike the space-time code design approach); hence, conventional channel codes can be used while the effects of spatial interference are addressed primarily in the signal processor design.

Foshini discusses different layering schemes for the proposed Bell Laboratories Layered Space-Time (BLAST) architecture [1]. In the simplest variation, the code words are transmitted in horizontal layers. Their preferred scheme, however, involves the transmission of code words in diagonal layers. For the purposes of this paper, we generalize the notion of a *layer*, defining it as a section of the transmission resources array having the property that each symbol interval within the section is allocated to at most one antenna. This property ensures that all spatial interference experienced by the layer comes from outside the layer. A layer has the further structural property that a set of spatial and/or temporal cyclic shifts of the layer within the transmission resource array provides a partitioning of the transmission resource array. This allows for a simple repeated use of the layer pattern for transmission of multiple, independent coded streams.

Formally, a layer in an $n \times \ell$ transmission resource array may be identified by an indexing set $L \subset I_n \times I_\ell$ having the property that the t-th symbol interval on antenna a belongs to the layer if and only if $(a, t) \in L$. Then, our formal notion of a layer requires that, if $(a, t) \in L$ and $(a', t') \in L$, then either $t \neq t'$ or $a = a'$—*i.e.*, that a is a function of t.

4 Multi-User Detection for Space-Time Applications

In this section, we cast the problem of space-time signal processing as a multi-user detection problem and consider iterative multi-user detection algorithms based on the turbo processing principle [11]. We derive a set of iterative multi-user detection algorithms that allow trade-offs to be made between performance and complexity. For simplicity, a layered system with binary channel codes (n binary channel encoders coupled to n transmit antennas) and binary phase shift keying (BPSK) is assumed. Extension to nonbinary codes with higher order modulation and to the multi-layered architecture is straightforward.

A soft-input/soft-output (SISO) multi-user detector module provides joint soft-decision estimates of the n streams of data. Each of the detected streams is decoded by a separate SISO channel decoder associated with the corresponding component channel code. After each decoding iteration, the soft outputs from the channel decoders are used to refine the processing performed by the SISO multi-user detector. Note that, in the iterative receiver, each of the streams is independently interleaved to facilitate convergence. This is a key feature of the receiver that also influences channel code design.

The overall complexity of the iterative receiver is driven primarily by the algorithm used in the multi-user detector. Therefore, three SISO, multi-user detection algorithms that provide a trade-off between performance and complexity are developed. The first is based on the maximum a-posteriori (MAP) probability rule [8]; the second is based on the minimum mean square error (MMSE) criterion [12]; and the third can be viewed a suboptimal approximation of the iterative MMSE receiver.

In all cases, the derivations require an assumption of statistical independence of the spatial soft decision information. This assumption is sufficiently satisfied in practice by requiring that the transmissions from each of the antennas be independently interleaved.

I. Iterative MAP Receiver

In this case, the SISO multi-user detector makes joint decisions regarding the n transmission streams via iterative computation of symbol-by-symbol maximum a posteriori (MAP) statistics. Specifically, the soft decision statistic for c_t^i is the updated log-likelihood ratio given by

$$L_t^i = \log \frac{\sum_{\underline{c}_t \in \mathcal{C}^+} P\left(\underline{r}_t | \underline{c}_t\right) P\left(c_t^1, .., c_t^{i-1}, c_t^{i+1}, .., c_t^n\right)}{\sum_{\underline{c}_t \in \mathcal{C}^-} P\left(\underline{r}_t | \underline{c}_t\right) P\left(c_t^1, .., c_t^{i-1}, c_t^{i+1}, .., c_t^n\right)} \tag{3}$$

where $P\left(\underline{r}_t | \underline{c}_t\right)$ is the conditional multivariate complex Gaussian distribution of the received vector; $P\left(c_t^1, ..., c_t^{i-1}, c_t^{i+1}, ..., c_t^n\right)$ is the joint a-priori probability distribution of the transmitted symbols; and

$$\begin{aligned} \mathcal{C}^+ &= \left\{\left(c_t^1, ..., c_t^{i-1}, +1, c_t^{i+1}, ..., c_t^n\right) : c_t^j \in \{-1, 1\}\right\} \\ \mathcal{C}^- &= \left\{\left(c_t^1, ..., c_t^{i-1}, -1, c_t^{i+1}, ..., c_t^n\right) : c_t^j \in \{-1, 1\}\right\} \end{aligned}$$

The computation of the joint distribution $P(c_t^1, ..., c_t^{i-1}, c_t^{i+1}, ..., c_t^n)$ is intractable in general without further assumptions. If statistical independence is assumed, then

$$\begin{aligned} & P\left(c_t^1, \ldots, c_t^{i-1}, c_t^{i+1}, \ldots, c_t^n\right) = \\ & P\left(c_t^1\right) \cdots P\left(c_t^{i-1}\right) P\left(c_t^{i+1}\right) \cdots P\left(c_t^n\right) \end{aligned} \tag{4}$$

In the first iteration, one takes $P\left(c_t^j = 1\right) = P\left(c_t^j = -1\right) = 1/2$. In subsequent iterations, the a-priori probabilities are re-computed based on the previous iteration's extrinsic information [13], λ_t^j, corresponding to the symbol transmitted from the j-th antenna at time t:

$$P\left(c_t^j = 1\right) = \frac{e^{\lambda_t^j}}{1 + e^{\lambda_t^j}}. \tag{5}$$

Note that, while the fading is assumed independent for each transmit-receive antenna pair, the extrinsic information will in general be spatially correlated—thereby invalidating the independence assumption—unless the separate antenna transmissions are independently interleaved. When different interleaving is used for each antenna transmission, however, the independence assumption is reasonably well approximated. The MAP approach has been extensively studied in the literature [14] [8] [9] for code division multiple access (CDMA) systems.

For the iterative MAP decoder, the number of terms in each of the summations in (3) is 2^{n-1}. Hence, the complexity of the MAP detector per iteration is $\mathcal{O}\left(n2^n\right)$, and the overall complexity of the receiver, per iteration, is $\mathcal{O}\left(n\left[2^n + 2^\nu\right]\right)$.

II. Iterative MMSE Receiver

In this scheme, the multi-user detection module is based on the MMSE criterion and is adapted from the first author's work [10] [12] on iterative MMSE multiuser detectors for CDMA systems. In this case, the SISO multi-user detector has the block diagram shown in Figure 1. After each decoding iteration, the soft outputs are used to update the a-priori probabilities of the transmitted symbols. These updated probabilities are then used to calculate the MMSE filter feed-forward and feedback weights. The feedback connections represent the subtractive interference cancellation part of the receiver, while the feed-forward weights serve to suppress the residual interference. While the block diagram allows for iterative channel estimation, for the purpose of this paper, the channel parameters are assumed to be known at the receiver.

The equations describing the filter coefficients used for generating the soft decision statistic corresponding to c_t^i will be derived. The subscript t will be omitted for convenience. Hence, the MMSE estimate y^i of the i-th antenna symbol at time t is given by

$$y^{(i)} = \underline{w}_f^{(i)T} \underline{r} + w_b^{(i)} \tag{6}$$

where $\underline{w}_f^{(i)}$ is the $m \times 1$ optimized feed-forward coefficients vector; $\underline{w}_b^{(i)}$ is a single coefficient that represents the soft cancellation part. Coefficients $\underline{w}_f^{(i)}$, $w_b^{(i)}$ are obtained by minimizing the mean square value of the error (e) between the data symbol and its estimate, given by

$$\begin{aligned} e &= E\left[\left|y^{(i)} - c^{(i)}\right|^2\right] \\ &= E\left[\left|\underline{w}_f^{(i)T} \underline{r} + w_b^{(i)} - c^{(i)}\right|^2\right] \end{aligned} \tag{7}$$

Now let $\underline{S}^{(i)}$ be the $m \times 1$ complex signature vector of the i^{th} transmit antenna; $S^{(n/i)}$ be the $m \times (n-1)$ matrix composed of the complex signature vectors of the other $n-1$ transmit antennas; $\underline{c}^{(n/i)}$ be the $(n-1) \times 1$ transmitted data vector form the other $n-1$ transmit antennas. Using standard minimization techniques, it is easily shown that the MMSE solutions for $\underline{w}_f^{(i)}$, and $w_b^{(i)}$ satisfy the following relations:

$$\underline{w}_f^{(i)^T} S^{(n/i)} E\left[\underline{c}^{(n/i)}\right] + w_b^{(i)} = 0 \tag{8}$$

$$\begin{aligned}\underline{w}_f^{(i)^T} &\left\{ \underline{S}^{(i)} \underline{S}^{(i)^H} + S^{(n/i)} E\left[\underline{c}^{(n/i)} \underline{c}^{(n/i)^T}\right] S^{(n/i)^H} \right\} \\ &+ \underline{w}_f^{(i)^T} E\left[\underline{n}\underline{n}^H\right] + S^{(n/i)} E\left[\underline{c}^{(n/i)}\right] w_b^{(i)} = \underline{S}^{(i)^H}\end{aligned} \tag{9}$$

where

$$E\left[\underline{n}\underline{n}^H\right] = N_0 I_{m\times m} \tag{10}$$

$$E\left[\underline{c}^{(n/i)}\right] = \tilde{\underline{c}}^{(n/i)}. \tag{11}$$

Assuming statistical independence assumption once again due to interleaving, we have

$$\begin{aligned}E\left[\underline{c}^{(n/i)} \underline{c}^{(n/i)^T}\right] &= I_{(n-1)\times(n-1)} \\ -Diag\left(\tilde{\underline{c}}^{(n/i)} \tilde{\underline{c}}^{(n/i)T}\right) &+ \tilde{\underline{c}}^{(n/i)} \tilde{\underline{c}}^{(n/i)T}.\end{aligned} \tag{12}$$

Here $I_{m\times m}$ is the identity matrix of order m; $\tilde{\underline{c}}^{(n/i)}$ is the $(n-1) \times 1$ vector of the expected values of the transmitted symbols from the other $n-1$ antennas. The a-priori probabilities used to evaluate these expected values are obtained from the previous decoding iteration's soft outputs via the component-wise relation (5).

To simplify notation, we define the following:

$$A = \underline{S}^{(i)} \underline{S}^{(i)^H} \tag{13}$$

$$\begin{aligned}B = S^{(n/i)} S^{(n/i)^H} &+ S^{(n/i)} \tilde{\underline{c}}^{(n/i)} \tilde{\underline{c}}^{(n/i)T} S^{(n/i)^H} \\ &- S^{(n/i)} Diag\left(\tilde{\underline{c}}^{(n/i)} \tilde{\underline{c}}^{(n/i)T}\right) S^{(n/i)^H}\end{aligned} \tag{14}$$

$$F = S^{(n/i)} \tilde{\underline{c}}^{(n/i)} \tag{15}$$

$$R_n = N_0 I_{m\times m} \tag{16}$$

Solving (8) and (9) for the optimum filter feed-forward and feedback coefficients gives

$$\underline{w}_f^{(i)^T} = \underline{S}^{(i)^H} \left(A + B + R_n - FF^H\right)^{-1} \tag{17}$$

and

$$w_b^{(i)} = -\underline{w}_f^{(i)^T} F. \tag{18}$$

The log-likelihood ratio is then given by

$$L^{(i)} = Re\left(w_f^{(i)^T} \underline{r} + w_b^{(i)}\right). \tag{19}$$

In the first decoding iteration, the transmitted symbols are assumed to have a uniform distribution; hence, $\tilde{\underline{c}}^{(n/i)} = \underline{0}$. The feed-forward filter coefficients vector, $\underline{w}_f^{(i)}$, in this iteration is given by similar relations to the MMSE equations derived in [15][1], and the feedback coefficient $w_b^{(i)} = 0$. After each iteration, $\tilde{\underline{c}}^{(n/i)}$ are recalculated using the decoders soft outputs. $\tilde{\underline{c}}^{(n/i)}$ are then used to generate the new set of filter coefficients as described. In the asymptotic case, when $\left|\tilde{\underline{c}}^{(n/i)}\right| = \underline{1}$, the receiver is equivalent to the subtractive interference canceler. This is expected, since $\left|\tilde{\underline{c}}^{(n/i)}\right| = \underline{1}$ means that the previous iteration decisions, for the other antenna symbols, are error free. Under this assumption, the subtractive interference canceler becomes the optimum solution.

III. Iterative Soft Interference Cancellation

The main source of complexity in the iterative MMSE approach is the matrix inversion operation required to compute the filter feed-forward coefficients (17). This observation motivates the following suboptimal approach. The MMSE estimate $y^{(i)}$ can be rewritten as

$$y^{(i)} = w_f^{(i)^T} \left[\underline{r} - S^{(n/i)} \tilde{\underline{c}}^{(n/i)}\right]. \tag{20}$$

---

[1] The only difference is that the relations in [15] were in the real domain while in our case the relations are in the complex domain because of the complex spreading codes.

Then, if the matched filter $w_f^{(i)^T} = [\alpha_{i1}^*, \ldots, \alpha_{im}^*]$ is used instead of the optimum weighting vector, the need for the matrix inversion operation in (17) is eliminated. The resulting receiver will have a linear complexity, per iteration, in the number of transmit antennas. A similar receiver, with a different derivation, was proposed for CDMA applications in [9].

4.0.1 Trade-Offs

The iterative MAP receiver offers a substantial reduction in complexity compared to the optimal receiver, but its complexity is still exponential and could be prohibitive for systems with medium to large numbers of antennas. The iterative MMSE receiver, however, has polynomial complexity. The soft interference cancellation method is the least complex.

The iterative MMSE receiver has an important advantage over the other two iterative approaches in that it can suppress the interference from other space-time users without the need to decode all of the signals. In the iterative MAP and soft interference cancellation techniques, undecoded signals are treated as white Gaussian noise. Hence, both approaches will suffer from a near-far problem from undecoded space-time users. In the iterative MMSE, however, the other users' interference can be suppressed by the feed-forward filter coefficients without the need to actually decode the other users' signals. The only requirement is prior knowledge of the other users' spreading codes, i.e. path gains.

5 Threaded Space-Time Architecture

In this section, we present a generic approach for space-time transmitter/receiver design well suited for iterative multi-user processing. In the proposed approach, the input data stream is divided into multiple threads. Each thread is encoded and interleaved separately. At each point of time, only one symbol is transmitted from each thread. At the receiver, the iterative multi-user detector serves to separate the different threads with minimal loss in performance. The encoding, interleaving, and distribution of thread symbols among different antennas is optimized to maximize spatial diversity and coding gain at the receiver, assuming no interference from the other threads. Meanwhile, interleaving is performed in such a way to maximize the efficiency of the iterative receiver.

As in the generic layered architecture, the transmitter has available a disjoint set of layers, $\mathcal{L} = \{L_1, L_2, \ldots, L_n\}$. The layer set $\mathcal{L}$ is designed so that each layer is active during all of the available symbol transmission intervals and, over time, uses each of the n antennas equally often. Thus, during each symbol transmission interval, the layers each transmit a symbol using a different antenna; and, in terms of antenna usage, all of the layers are equivalent. A layer satisfying these constraints is referred to as a *thread* of spatial span n. The simplest example of threaded layering is the set $\mathcal{L}$ in which

$$L_i = \{((t+i) \bmod n + 1, t) : 0 \leq t < N\}. \quad (21)$$

where N is the temporal span of the layer in symbols. Unlike the layered architectures of [1], the new design approach treats the coded transmission in each layer as a bona fide space-time code, constructions for which are given in [16] [17]. Looking at the space-time coding performed on a single layer in isolation, one notes that the main limitation of this construction is the reduction in throughput resulting from the silence periods imposed on the different antennas. But, in the overall threaded transmission scheme, the silent periods on antennas not used by a given layer are filled with the transmissions from the other component space-time codes. Advanced signal processing at the receiver, necessary to remove or suppress spatial interference among the threaded layers, allows high throughput to be achieved. One innovation of the new architecture is that, under the assumption of error free interference cancellation, the component space-time codes can be designed to achieve full spatial diversity without degradation in overall system throughput.

The new space-time architecture is not a multi-layered approach since the transmit positions occupied by the modulated code symbols for a particular thread constitute a single layer. Yet, the new architecture is not a layered architecture in the same sense as the BLAST architecture, since the threaded layering is a more general type of layering well-suited for iterative multi-user techniques and the channel coding design in the new approach is two-dimensional based on space-time coding principles designed to exploit both the spatial and temporal diversity. To distinguish this new approach, we refer to it as the *Threaded Space-Time* (TST) architecture. In [17] [16], the authors developed an algebraic framework for threaded space-time code design. This framework encompasses trellis and block codes for arbitrary number of antennas with arbitrary complex constellations.

It is clear that the efficiency of the threaded architecture depends heavily on the ability of the receiver to eliminate the interference coming from the other space-time component codes. In principle, any multi-user detection technique can be used in this context. We will focus our attention on the iterative MMSE receiver because of its reasonable complexity and its ability to achieve performance close to the interference-free scenario under different conditions.

6 Performance Comparisons

In this section, the different schemes are compared via simulation. Throughout our simulation study, we used the rate 1/2, 4-state convolutional code with generator polynomial $(5_8, 7_8)$. The main advantage of convolutional codes is the availability of computationally efficient soft-input-soft-output decoders. Periodic bit demultiplexers were used to distribute the encoder outputs across the different antennas. In addition, inner random interleavers were used to allow for better convergence characteristics for the iterative MMSE receiver. The results were obtained by averaging the frame error rates of all the component codes. The channel decoder was based on the soft output Viterbi algorithm (SOVA). The frame length was 100 bits. The channel was assumed to follow the quasi-static fading model.

Figure 2 compares the performance of the iterative MMSE receiver with horizontal layering and the interference-free performance which assumes the presence of only one transmit antenna and 4 receive antennas. For the iterative MMSE, the number of transmit and receive antennas is 4 and the bandwidth efficiency is $\eta = 2$ bits/sec/hz (i.e, BPSK modulation). In general, the relation between the energy per bit to noise ratio E_b/N_0 and the transmitted signal to noise ratio is

$$SNR = \eta \frac{E_b}{N_0} \tag{22}$$

The interference-free scenario represents a lower bound on the performance achieved by the optimum receiver. It is shown that the iterative MMSE performs within a fraction of a dB from that lower bound.

Now, we are interested in comparing the performance of the threaded space-time architecture (TST) presented in Section 5 with the LST and the multi-layereing architecture proposed by Tarokh, Naguib, Seshadri, and Calderbank (TNSC) in [2].

Figure 3 compares the TST with a lower bound on the frame error rate achieved by the diagonal LST. This lower bound assumes error free decision feedback. In practice, the performance of the LST is expected to be close to the lower bound at high signal-to noise ratios and the bound will become looser as the signal-to-noise ratio decreases. The iterative MMSE receiver is shown to provide a 3 dB gain over the diagonal LST lower bound under these conditions.

Figures 4 and 5 compare the performance of the TST and TNSC architectures for the cases of 4 transmit/4 receive and 8 transmit/8 receive antennas, respectively. QPSK modulation with Gray mapping was used to map the binary input at each antenna to a complex constellation. Hence, the spectral efficiency is 4 bits/sec/Hz and 8 bits/sec/Hz, respectively. The results of the TNSC scheme are obtained from Figures 4 and 6 in [2], respectively. From the figures, the significant gain provided by the TST over the TNSC scheme is clear. Indeed, the TST approach shows a gain of 4–8 dB over the TNSC scheme. The TST results are within 2–3 dB of the outage capacity. The gain in diversity advantage achieved by the TST can be seen in the steeper asymptotic slope of the performance curve. It is also shown that the gain provided by the TST increases with the number of antennas. This can be attributed to the better exploitation of the diversity in the TST.

7 Conclusions

In this paper, we took a fresh look at the design problem for multiple antenna systems operating over a fading channel. The problem was addressed from both a signal processing and a space-time coding perspective. On the signal processing side, we presented a set of iterative algorithms for joint decoding and detection that provide a trade-off between performance and complexity. Simulation results were provided for the iterative MMSE receiver establishing its ability to approach the interference-free performance lower bound within a fraction of a dB. From the space-time coding perspective, we proposed the new threaded space-time architecture that combines space-time coding with iterative multi-user detection. The new approach was shown, through simulation, to achieve significant gains over the combined array processing and space-time coding recently published by Tarokh *et al.* [2].

References

[1] G. J. Foschini. Layered space-time architecture for wireless communication in fading environments when using multiple antennas. *Bell Labs Tech. J.*, 2, Autumn 1996.

[2] V. Tarokh, A. Naguib, N. Seshadri, and A. R. Calderbank. Combined array processing and space-time coding. *IEEE Trans. Info. Theory*, IT-45:1121–1128, May 1999.

[3] E. Teletar. Capacity of multi-antenna gaussian channels. *Technical Report, AT&T-Bell Labs*, June 1995.

[4] G. J. Foschini and M. Gans. On the limits of wireless communication in a fading environment when using multiple antennas. *Wireless Personal Communication*, 6:311–335, Mar 1998.

[5] V. Tarokh, N. Seshadri, and A. R. Calderbank. Space-time codes for high data rate wireless communication: Performance criterion and code construction. *IEEE Trans. Info. Theory*, IT-44:774–765, March 1998.

[6] J.-C. Guey, M. R. Bell M. P. Fitz, and W.-Y. Kuo. Signal design for transmitter diversity wireless communication systems over Rayleigh fading channels. *IEEE Vehicular Technology Conference*, pages 136–140, Atlanta, 1996.

[7] A. R. Hammons Jr. and H. El Gamal. On the theory of space-time codes for PSK modulation. *to appear in IEEE Trans. Info. Theory.*

[8] M. C. Reed, C. B. Schlegel, P. D. Alexander, and J. A. Asenstorfer. Iterative multiuser detection for DS-CDMA with FEC. *International Symposium on Turbo Codes and Related Topics*, Brest-France:162–165, Sept 1997.

[9] P. D. Alexander, A. J. Grant, and M. C. Reed. Iterative detection in code division multiple access with error control coding. *European Transactions on Telecommunications*, 9:419–425, Sep/Oct 1998.

[10] H. El Gamal. On the theory and applications of space-time and graph based codes. *Ph.D Dissertation, University of Maryland at College Park*, May 1999.

[11] J. Hagenauer. The turbo principle: Tutorial introduction and state of the art. *International Symposium on Turbo Codes and Related Topics*, Brest-France:1–9, Sept 1997.

[12] H. El Gamal and E. Geraniotis. Iterative multi-user detection for coded CDMA signals in AWGN and Rayleigh fading channels. *to appear IEEE Journal on Selected Areas in Communications*, Oct 1999.

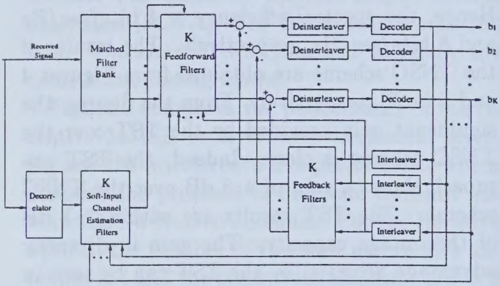
[13] S. Benedetto, G. Montorsi, D. Divsalar, and F. Pollara. A soft-input soft-output maximum a posteriori module to decode parallel and serially concatenated codes. *Telecom. and Data Acquisition Progress Reports, Jet Propulsion Laboratory*, 42-127, Nov 1996.

[14] M. Moher. An iterative multiuser decoder for near-capacity communications. *IEEE Transactions on Communications*, 46:870–880, July 1998.

[15] U. Madhow and M. Honig. MMSE interference suppression for direct sequence spread spectrum CDMA. *IEEE Transactions on Communications*, 42:3178–3188, December 1994.

[16] A. R. Hammons and H. El Gamal. Further results on the algebraic design of space-time codes. *submitted to ISIT 2000*, Sept 1999.

[17] H. El Gamal and A. R. Hammons Jr. The layered space-time architecture: a new prespective. *submitted to IEEE Trans. Info. Theory*, 1999.



MMSE multiuser detector and iterative decoding

Figure 1. Block Diagram of the Iterative MMSE Receiver for Space-Time Applications with K transmit antennas.

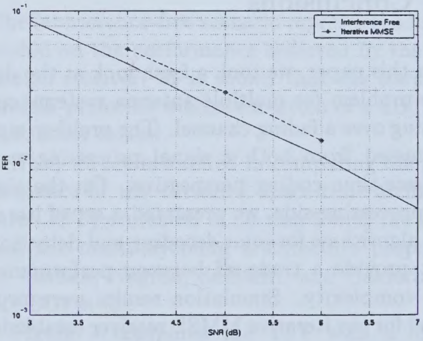


Figure 2. Performance of the iterative MMSE receiver.

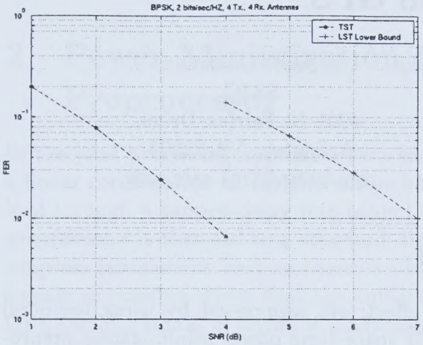


Figure 3. Performance of the TST and LST archietctures .

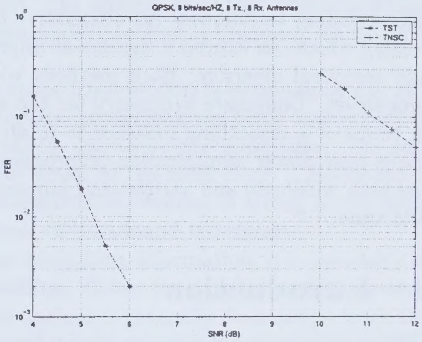


Figure 5. Performance of the TST and TNSC architectures.

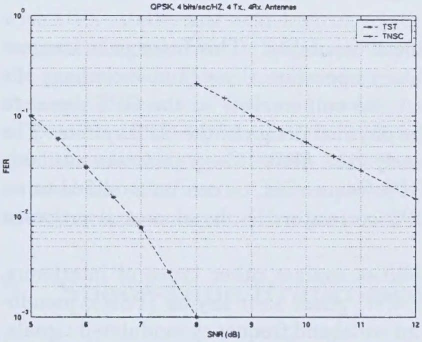


Figure 4. Performance of the TST and TNSC archietctures

Interference Suppression for CDMA via a Space-Time Power Minimization Based Preprocessor with Applications to GPS[1]

Wilbur L. Myrick & Michael D. Zoltowski
School of Electrical Engineering
Purdue University
West Lafayette, IN 47907-1285
e-mail: [wlm, mikedz]@ecn.purdue.edu

J. Scott Goldstein
SAIC
4001 N. Fairfax Drive
Arlington, VA 22203
sgoldstein@trg1.saic.com

Abstract

Narrowband jamming in a CDMA system can be effectively suppressed by using temporal preprocessing techniques. An alternative technique is space-time power minimization preprocessing which can exhibit a significant improvement in receiver performance by utilizing spatial and temporal diversity. Both methods will cause some PN code distortion due to temporal processing. Conjugate symmetry for the space-time preprocessor is developed and presented in terms of a smoothing measure to minimize PN code distortion. It is demonstrated that a space-time power minimization preprocessor with conjugate symmetry is capable of suppressing both wideband and narrowband jammers while provided enhanced smoothness across space and time.

1 Introduction

GPS is known to provide significant force enhancement capability. This force enhancement capability has been demonstrated in every U.S. military operation since (and including) the Gulf War, but with this capability is a concern about the vulnerability of the GPS signal to jamming. The jamming threat is serious because of the physical design of the GPS system. The received power from the GPS satellites is approximately -157 dBW. Many jammers available on the arms market today either already cover the GPS frequencies, or can be modified to do so. A space-time preprocessing filter prior to the GPS correlators is one of several proposed methods for suppressing jammers.

The filter should also provide suppression capabilities against other types of interferers, including continuous wave (CW), swept CW, pulsed CW, phase shift keying (PSK), pseudonoise signals (20 MHz bandwidth), and narrowband and wideband frequency modulated signals. The interferers may be located anywhere within or adjacent to the 20.46 MHz bandwidths centered at the GPS L1 frequency of 1575.42 MHz or the L2 frequency of 1227.60 MHz. The interferers may be distributed anywhere over 2π steradians of solid angle centered at zenith relative to the local horizontal plane of a GPS receiving antenna. The number of interferers may be as high as twenty or more.

In addition to providing interference suppression, the filter should allow reception of GPS satellite signals in a stressed environment by maximizing the signal-to-interference plus noise

---

[1] This research was supported by the Air Force Office of Scientific Research under contract no. F49620-97-1-0275.

ratio (SINR) while minimizing distortion of the desired GPS signals for acquisition and tracking. This filter should furthermore be targeted for operation within the environmental requirements of the dynamics which characterize a high performance fighter aircraft.

2 Power Minimization Based Joint Space-Time Preprocessor

In the joint processing approach, each sample value input to the GPS receiver is formed from a linear combination of samples across both space and time. The space-time weights are realized through a tapped-delay line behind each digitized baseband antenna. The output of the preprocessor is then fed to a standard digital GPS receiver. The goal of the preprocessor is to suppress jammers as best as possible while simultaneously passing as many undistorted GPS signals as possible. Note that the anti-jam space-time filter will not be optimized for any one GPS satellite signal in terms of maximizing the SINR. The advantage of this approach is that the anti-jam space-time filter remains a separate component so that a standard digital GPS receiver may be employed.

The criterion for determining the optimal set of space-time weights is premised on the fact that the respective power levels of the desired GPS signals are significantly below the noise floor and that the jammers that could have deleterious effects are above the noise floor. The goal then is to drive the power of the preprocessor output down to the noise floor. This approach serves to place point nulls at the respective angle-frequency coordinates of strong narrowband interferers and spatial nulls in the respective directions of broadband interferers.

In order for the GPS receiver to provide accurate navigation information, it is necessary to track the signals from at least four different GPS satellites. Given the parallax error associated with GPS satellites at near-horizon relative to the aircraft, it is generally desirable to track the respective signals from a larger number of GPS satellites, e.g., twelve. It is desired then that the preprocessor "pass" unaltered as many GPS signals as possible. Thus, the magnitude of the multidimensional Fourier transform of the space-time weights should be as flat (smooth) as possible in the spectral domain as a function of frequency and angular dimensions. The goal then is achieve a desired smoothness while simultaneously nulling both wideband and narrowband interferers. This motivates the use of an objective function that measures the smoothness associated with a particular set of space-time weights.

3 Space-time Preprocessing and Filter Distortion Effects

It is known that a tapped-delay-line preprocessor of a spread spectrum signal (such as the GPS signal) introduces self-noise and some distortion on the desired GPS signal. Characterization of this type of distortion in temporal preprocessing has previously been studied by [2]. Our work extends their results by considering a space-time implementation of such an interference suppression algorithm to effectively null both wideband and narrowband jammers.

A space-time power minimization based preprocessor for GPS was proposed in [4] for anti-jam protection. The output power of the space-time preprocessor is minimized under the constraint that the value of the first tap of the tapped-delay-line behind the reference element be unity. This is not necessarily the optimum constraint to reduce GPS signal distortion while nulling out the wideband and narrowband jammers. An alternative approach is taken in this paper in which the space-time weights are constrained to be conjugate symmetric. The effects of choosing conjugate symmetric weights versus non-symmetric weights constitutes the need to develop a smoothness measure. The smoothness function, introduced in [5], is a measure of the deviation from flatness of the magnitude of the 2D FFT of the space-time weight vector.

The 2D FFT of a space-time weight vector **h** can be expressed in terms of a spatial frequency, μ, and an angular digital frequency, ω, which are defined as follows. First, for sake of simplicity, we here assume a linear array of identical antennas equi-spaced by $d = \lambda/2$ along a line, where λ is the wavelength associated with the L1 frequency. In this scenario, the spatial frequency is defined as

$$\mu = 2\pi \frac{d}{\lambda} \sin\theta = \pi \sin\theta$$

where θ is the angle-of-arrival (AOA). The angular digital frequency is defined as $\omega = 2\pi\frac{\Delta f}{F_s}$ where Δf is the frequency offset relative to the L1 frequency and F_s is the sampling rate.

With these definitions, the 2D FFT of **h** may be expressed as

$$H(\mu,\omega) = \mathbf{h}^H \mathbf{f}(\mu,\omega) \tag{1}$$

where $\mathbf{f}(\mu,\omega) = \mathbf{f}_M(\mu) \otimes \mathbf{f}_N(\omega)$. Now $\mathbf{f}_M(\mu)$ and $\mathbf{f}_N(\omega)$ are respectively defined as

$$\mathbf{f}_M(\mu) = [1, e^{j\mu}, e^{j2\mu}, ..., e^{j(M-1)\mu}]^T \tag{2}$$

$$\mathbf{f}_N(\omega) = [1, e^{j\omega}, e^{j2\omega}, ..., e^{j(N-1)\omega}]^T. \tag{3}$$

We desire the space-time weight vector to place a null at the angle-frequency coordinate (μ_i, ω_i) of the i-th narrowband jammer

$$\mathbf{h}^H \mathbf{f}(\mu_i, \omega_i) = 0 \tag{4}$$

and a spatial null in the direction of the ℓ-th wideband jammer

$$\mathbf{h}^H \mathbf{f}(\mu_\ell, \omega) = 0. \tag{5}$$

We seek now to extend this result with that of conjugate symmetry. The smoothness function is defined as

$$f(\mathbf{h}) = \frac{1}{4\pi^2} \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} \left| e^{jm\mu} e^{jn\omega} - \mathbf{h}^H \mathbf{f}(\mu,\omega) \right|^2 d\mu d\omega. \tag{6}$$

This function is used as a measure of smoothness associated with a particular **h**. The smaller the value of $f(\mathbf{h})$, the smoother the 2D FFT spectrum of **h**. Equation (6) can be rewritten as

$$\frac{1}{4\pi^2} \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} \Big\{ 1 - 2\mathbf{Re}\Big\{ e^{jm\mu} e^{jn\omega} \mathbf{h}^H \mathbf{f}(\mu,\omega) \Big\} + \mathbf{h}^H \mathbf{f}(\mu,\omega) \mathbf{f}^H(\mu,\omega) \mathbf{h} \Big\} d\mu d\omega. \tag{7}$$

Distributing the double integral yields

$$\frac{1}{4\pi^2} \Big\{ 4\pi^2 - 2\mathbf{Re}\Big\{ \mathbf{h}^H \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} e^{jm\mu} e^{jn\omega} \mathbf{f}(\mu,\omega) d\mu d\omega \Big\} + \mathbf{h}^H \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} \mathbf{f}(\mu,\omega) \mathbf{f}^H(\mu,\omega) d\mu d\omega \mathbf{h} \Big\}. \tag{8}$$

Using previous definitions, Kronecker Delta product properties, and integral separation, it can be shown that

$$\int_{-\pi}^{\pi} \int_{-\pi}^{\pi} e^{jm\mu} e^{jn\omega} \mathbf{f}(\mu,\omega) d\mu d\omega = \int_{-\pi}^{\pi} e^{jn\omega} \mathbf{f}_N(\omega) d\omega \otimes \int_{-\pi}^{\pi} e^{jm\mu} \mathbf{f}_M(\mu) d\mu = 4\pi^2 [\boldsymbol{\delta}_n \otimes \boldsymbol{\delta}_m]. \tag{9}$$

and

$$\int_{-\pi}^{\pi}\int_{-\pi}^{\pi}\mathbf{f}(\mu,\omega)\mathbf{f}^{H}(\mu,\omega)d\mu d\omega=\int_{-\pi}^{\pi}\int_{-\pi}^{\pi}\mathbf{f}_{M}(\mu)\mathbf{f}_{M}^{H}(\mu)\otimes\mathbf{f}_{N}(\omega)\mathbf{f}_{N}^{H}(\omega)d\mu d\omega=4\pi^{2}\mathbf{I}. \tag{10}$$

where $\boldsymbol{\delta}_n = [\delta(n), ..., \delta(n-(N-1))]$ and $\boldsymbol{\delta}_m = [\delta(m), ..., \delta(m-(M-1))]$. After making the proper substitutions into (8), our smoothing function simplifies to

$$f(\mathbf{h}) = 1 - 2\mathbf{Re}\left\{\mathbf{h}^H[\boldsymbol{\delta}_n \otimes \boldsymbol{\delta}_m]\right\} + \mathbf{h}^H\mathbf{h} \tag{11}$$

In our simulations, we always choose n and m such that the unity weight of $\mathbf{h}$ is selected thereby simplifying $2\mathbf{Re}\left\{\mathbf{h}^H[\boldsymbol{\delta}_n \otimes \boldsymbol{\delta}_m]\right\} = 2$ and (11) becomes

$$f(\mathbf{h}) = -1 + \mathbf{h}^H\mathbf{h} \tag{12}$$

Equation (12) is a measure of smoothness associated with a given $\mathbf{h}$ having a unity weight constraint, i.e., with one of the elements of $\mathbf{h}$ constrained to be unity. Thus, under a unity weight constraint on a particular element of the space-time weight vector, $\mathbf{h}$, equation (12) implies that the smaller the norm of $\mathbf{h}$, the better the smoothness. Simulation studies have revealed that a space-time weight vector exhibiting conjugate centro-symmetry yields a lower deviation from smoothness than space-time weight vectors that do not exhibit conjugate centro-symmetry. Note that conjugate centro-symmetry can only be realized by placing the unity weight constraint on the center tap behind the center antenna. Before discussing the simulations, it is necessary to illustrate what is meant by conjugate symmetry in terms of space-time processing.

3.1 Conjugate Symmetry

Conjugate symmetry may be imposed on the space-time weights by forward-backward averaging the space-time correlation matrix and selecting the unity weight tap to be the $\frac{(N+1)}{2}$-th tap behind the $\frac{(M+1)}{2}$-th antenna assuming N and M to be odd. To formally prove this let us define the forward-backward space-time correlation matrix as

$$\mathbf{K}_{fb} = \frac{1}{2}\left\{\mathbf{K} + \tilde{\mathbf{J}}\mathbf{K}^{*}\tilde{\mathbf{J}}\right\}. \tag{13}$$

where $\tilde{\mathbf{J}}$ is an $(NM \times NM)$ diagonal matrix having 1's on the anti-diagonal from bottom left to top right. Note that $\mathbf{K}_{fb}$ satisfies the following property

$$\tilde{\mathbf{J}}\mathbf{K}_{fb}\tilde{\mathbf{J}} = \mathbf{K}_{fb}^{*} \tag{14}$$

This property is easily proved using the property $\tilde{\mathbf{J}}\tilde{\mathbf{J}} = \mathbf{I}$.

$$\tilde{\mathbf{J}}\mathbf{K}_{fb}\tilde{\mathbf{J}} = \frac{1}{2}\left\{\tilde{\mathbf{J}}\mathbf{K}\tilde{\mathbf{J}} + \tilde{\mathbf{J}}\tilde{\mathbf{J}}\mathbf{K}^{*}\tilde{\mathbf{J}}\tilde{\mathbf{J}}\right\} = \mathbf{K}_{fb}^{*}$$

Substituting $\mathbf{K}_{fb}$ for $\mathbf{K}$, the power minimization problem becomes

$$\mathbf{h}^H\mathbf{K}_{fb}\mathbf{h} \quad \text{subject to: } \mathbf{h}^H\boldsymbol{\delta}_{\frac{(NM+1)}{2}} = 1 \tag{15}$$

where $\boldsymbol{\delta}_{\frac{(NM+1)}{2}}$ is the $NM \times 1$ vector $\boldsymbol{\delta}_{\frac{(NM+1)}{2}} = [0, 0, .., 1, .., 0]^T$ where the 1 is located in the $\frac{NM+1}{2}$ position of the vector. Using the method of Lagrange multipliers, the solution to (15) may be found by solving

$$\mathbf{K}_{fb}\mathbf{h} = \alpha\boldsymbol{\delta}_{\frac{(NM+1)}{2}} \tag{16}$$

where α is the Lagrange multiplier used to satisfy the unity weight constraint $\mathbf{h}^H \boldsymbol{\delta}_{\frac{(NM+1)}{2}} = 1$. It is easily shown that

$$\alpha = \frac{1}{\boldsymbol{\delta}^H_{\frac{(NM+1)}{2}} \mathbf{K}_{fb}^{-1} \boldsymbol{\delta}_{\frac{(NM+1)}{2}}} \tag{17}$$

Since $\mathbf{K}_{fb}$ is Hermitian-symmetric, it follows that $\mathbf{K}_{fb}^{-1}$ is Hermitian-symmetric, so that α is real-valued.

Using the property $\tilde{\mathbf{J}}\tilde{\mathbf{J}} = \mathbf{I}$ and premultiplying both sides of (16) by $\tilde{\mathbf{J}}$ yields

$$\tilde{\mathbf{J}}\mathbf{K}_{fb}\tilde{\mathbf{J}}\tilde{\mathbf{J}}\mathbf{h} = \alpha \tilde{\mathbf{J}} \boldsymbol{\delta}_{\frac{(NM+1)}{2}} \tag{18}$$

Using the property $\tilde{\mathbf{J}}\mathbf{K}_{fb}\tilde{\mathbf{J}} = \mathbf{K}_{fb}^*$, (18) can be rewritten as

$$\mathbf{K}_{fb}^* \tilde{\mathbf{J}}\mathbf{h} = \alpha \tilde{\mathbf{J}} \boldsymbol{\delta}_{\frac{(NM+1)}{2}} \tag{19}$$

Conjugating both sides of (19) yields

$$\mathbf{K}_{fb} \tilde{\mathbf{J}}\mathbf{h}^* = \alpha \tilde{\mathbf{J}} \boldsymbol{\delta}_{\frac{(NM+1)}{2}} = \alpha \boldsymbol{\delta}_{\frac{(NM+1)}{2}} \tag{20}$$

Simplifying

$$\mathbf{K}_{fb} \{\tilde{\mathbf{J}}\mathbf{h}^*\} = \alpha \boldsymbol{\delta}_{\frac{(NM+1)}{2}} \tag{21}$$

Recall that $\mathbf{h}$ also satisfies $\mathbf{K}_{fb}\mathbf{h} = \alpha \boldsymbol{\delta}_{\frac{(NM+1)}{2}}$. Since $\mathbf{K}_{fb}$ is full rank, it follows that the solution is unique. In turn, this implies

$$\tilde{\mathbf{J}}\mathbf{h}^* = \mathbf{h} \tag{22}$$

This indicates that the solution $\mathbf{h}$ exhibits conjugate centro-symmetry. This property only holds if forward-backward averaging is employed and the unity weight is placed at the center element of $\mathbf{h}$. It is important to note that backward-forward averaging effectively doubles the snapshots since for every $\mathbf{x}[n]$, there is another snapshot $\tilde{\mathbf{J}}\mathbf{x}^*[n]$. In addition, enforcing conjugate centro-symmetry should also yield faster convergence since the number of weights to be updated is effectively halved.

4 Simulations

In the simulations $M = N = 7$. These definitions imply a $M = 7$ element equi-spaced linear array with $N = 7$ taps at each antenna. Table 1 summarizes values of the smoothness function $f(\mathbf{h})$ for various cases of unity weight location, number of snapshots, and use of forward-backward averaging or not. In case (i) (1st tap, 1st antenna) there was no forward-backward averaging on the space-time correlation matrix and selects the 1st tap behind the 1st antenna to have unity weight. Case (ii) (4th tap, 4th antenna w/o FB) uses no forward-backward averaging, but instead selects the 4th tap behind the 4th antenna. This was based on our selection of $M = 7$ and $N = 7$. In general, one selects the $\frac{N+1}{2}$-th tap behind the $\frac{M+1}{2}$-th antenna to have unity weight. Case (iii) (4th tap, 4th antenna w/ FB) is the same as case (ii) except it uses forward-backward averaging of the space-time correlation matrix to guarantee $\mathbf{h}$ be conjugate centro-symmetric. A small number of jammers were selected to illustrate the effects of smoothing on the 2D FFT spectrum. Assuming a $20MHz$ receiver bandwidth at each antenna, the noise floor was determined to be at $\sim130dBW$ after filtering at each antenna.

Notice the forward-backward averaged space-time weights obtained in case (iii) have the least value for each of the three snapshot totals simulated. Observe that the $f(\mathbf{h})$ values obtained in case (iii) for a particular number of snapshots is less than the $f(\mathbf{h})$ values obtained in cases (i) and (ii) with twice the number of snapshots. Figures 1-6 illustrate the smoothness of the 2D FFT spectrum of $\mathbf{h}$ associated with cases (i), (ii), and (iii) with both 125 and 250 snapshots. Notice the contrast in smoothness associated with each selected position of the unity weight constraint and the effects of forward-backward averaging. While all cases provide the desired nulls for both the narrowband and wideband jammers, selecting the unity weight to be at the center element of the of the space-time weight vector yields a smoother 2D FFT spectrum across space and frequency versus other locations for the unity weight constraint. This was previously observed in [5]. We extend those results by illustrating that the conjugate symmetry constraint in addition to selecting the unity weight at the center element exhibits a "smoother" 2D FFT spectrum. Smoothness is here quantitatively expressed in terms of the smoothness function $f(\mathbf{h})$ defined in Equation (6).

Table 1: Simulation Parameters and $f(\mathbf{h})$ Values

Jammer Type	SNR	AOA	Bandwidth
Wideband	-110 dBW	$-60°$	20 MHz
Wideband	-110 dBW	$60°$	20 MHz
Narrowband	-100 dBW	$0°$	-8 MHz
Case Type	125 Snapshots	250 Snapshots	500 Snapshots
(i) 1st Tap w/o FB ; $f(\mathbf{h})$ =	1.3037	0.9045	0.6395
(ii) center tap w/o FB; $f(\mathbf{h})$ =	0.7630	0.6792	0.5875
(iii) center tap w FB; $f(\mathbf{h})$ =	0.6005	0.5486	0.5170

5 Conclusion

An objective function that measures smoothness for a space-time preprocessor was presented. By forward-backward averaging of the space-time correlation matrix, it was shown that the space-time weight vector that minimizes output power subject to a unity weight constraint on the center element exhibits conjugate centro-symmetry. The simulations revealed that the conjugate centro-symmetry space-time weight vectors yielded a lower value of the smoothness function; the better smoothness implies lower distortion to the GPS signals.

References

[1] D. H. Brandwood, "A complex gradient operator and its application in adaptive array theory," *IEE Proc.,Parts F and H,* vol. 130, pp. 11-16, Feb. 1983

[2] J. W. Ketchum and J. G. Proakis, "Adaptive Algorithms for Estimating and Suppressing Narrow-band Interference in PN Spread-Spectrum Systems," *IEEE Trans. on Communications,* May 1982, vol. COM-30, pp. 1169-1177.

[3] A. S. Gecan and M. D. Zoltowski, "Power Minimization Techniques for GPS Null Steering Antennas," *Institute of Navigation (ION) Conference,* Palm Springs, CA, 13-15 Sept. 1995.

[4] W. L. Myrick, M. D. Zoltowski and J. Scott Goldstein, "Anti-Jam Space-Time Preprocessor for GPS Based on Multistage Nested Wiener Filter," *Milcom '99.*

[5] W. L. Myrick, M. D. Zoltowski and J. Scott Goldstein, "Smoothing of Space-Time Power Minimization Based Preprocessor for GPS," *DASP '99.*

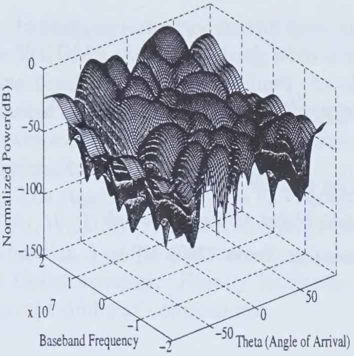


Figure 1. 1st tap, 1st antenna
(125 snapshots)

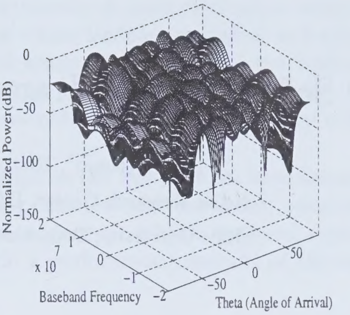


Figure 4. 1st tap, 1st antenna
(250 snapshots)

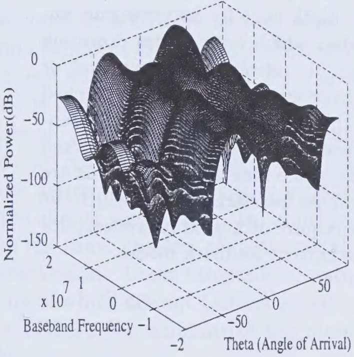


Figure 2. 4th tap, 4th antenna w/o FB
(125 snapshots)

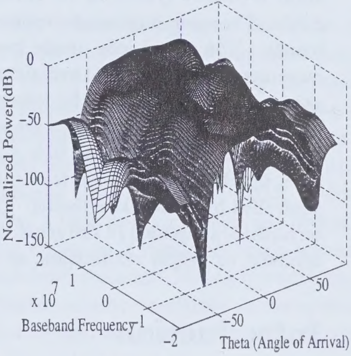


Figure 5. 4th tap, 4th antenna w/o FB
(250 snapshots)

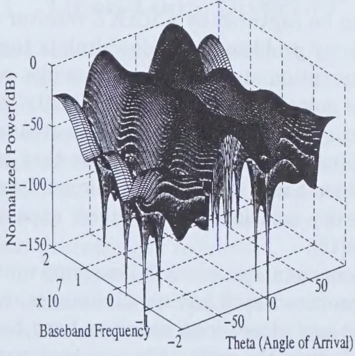


Figure 3. 4th tap, 4th antenna w/ FB
(125 snapshots)

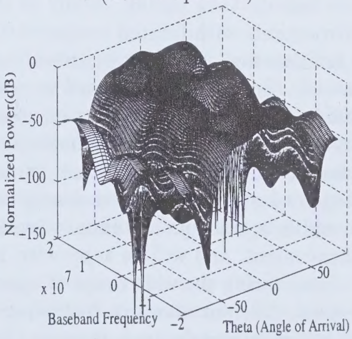


Figure 6. 4th tap, 4th antenna w/ FB
(250 snapshots)

Soft–Weighted Transmit Diversity for WCDMA

Ari Hottinen[1], Risto Wichman[1] and Dinesh Rajan[2]

[1] Nokia Research Center, P.O. Box 407 00045 Nokia Group, Finland

[2] Rice University, Department of ECE, Houston, Texas 77005 USA

Abstract

The transmit diversity concept adopted for the FDD mode of the third generation WCDMA system in 3G standardization has an open-loop and a closed-loop mode. The open-loop mode applies a space-time block code for two transmit antennas. The closed-loop mode has two sub-modes, which both utilize downlink measurements at terminal and subsequent feedback in controlling the phase and/or gain transmit weights in the transmit antennas in order to approximate matched beamforming. These concepts are summarized in this paper for two transmit antennas and they are compared to a *soft-weighted* transmit diversity concept, proposed here, in which the relative transmit powers of the space-time encoded signals are controlled by feedback signaling. This leads to a robust feedback mode in which the terminal is simplified in the sense that it does not require dedicated reference/pilot symbols, or weight verification, as opposed to the current feedback modes.

1 Introduction

Future wireless services like remote web browsing are expected to burden the downlink more heavily than uplink. Many of these services are likely to be used in low-mobility environments with limited temporal diversity (e.g to be captured by a RAKE receiver or by interleaving). The lack of diversity causes a capacity problem in the downlink in these channels. Space-diversity based on multi-antenna reception at the terminal provides one remedy to the problem. However, it is an expensive solution in terms of complexity and cost of the mobile handset. Instead, receive diversity, e.g. with two receive antennas, is seen as an add-on feature, not requiring standardization, and as a technique that can be applied at manufacturers discretion. On the other hand, diversity techniques based on multiple transmit antenna provide low complexity solutions for downlink capacity enhancement, and will be supported in 3G systems[18].

Considering the wide range of operating environments under which 3G systems would be used, different diversity techniques have been proposed each having advantages over the others in certain environments. Both open-loop and closed-loop concepts have been under consideration for the 3G WCDMA system. Many of the previously proposed open-loop techniques, see e.g. [17, 5, 3, 7] have been considered but the selected open-loop concept applies a two-dimensional space-time block code proposed by Alamouti [1]. The open-loop technique is applied when very little or nothing is known from the downlink channel at the base station. It is well known that additional performance improvements are achievable in closed-loop systems where the transmitter is provided with suitable side information on the downlink channel characteristics [9, 10, 12].

In this paper we review the open and closed-loop transmit diversity techniques adopted for WCDMA systems, along with a new concept in which feedback signaling and space-time block codes are combined. In all the feedback modes the capacity of the feedback channel is maintained at 1.5 kbps and this is used to achieve significant (3-6 dB) improvements over single-antenna transmission in low-mobility channels with only a few dominant multipath components.

The paper is structured as follows. Section 2 describes the key physical layer features of the WCDMA transmit diversity concept, including the open and closed-loop solutions. In Section 3 we propose the concept of a soft-weighted space-time block code and explain its characteristics. Finally in Section 4 we provide a performance analysis of the new concept and our conclusions.

2 WCDMA Transmit Diversity Concept

We consider a CDMA system with two synchronously transmitting antennas. The antenna elements are spaced sufficiently close to each other so that the propagation delays between each antenna and a given Mobile Station (MS) are approximately the same. This is important in order to maintain downlink orthogonality at least in a single-path channel. Also, we assume that the antennas are sufficiently far apart from each other in order to reduce channel amplitude correlation between the antennas. In practice, both of these assumptions are usually satisfied when the antenna spacing of $10\lambda - 20\lambda$, where λ is the carrier wavelength. In addition it is desired that a user applying transmit diversity uses the same number of (orthogonal) channelization codes as it would in single-antenna transmission. These targets lead naturally to symbol-level transmit diversity techniques, many of which are not CDMA specific, but are applicable in other multiple access systems like TDMA. We summarize the main characteristics of currently accepted 3G concepts [18].

2.1 Open-Loop Diversity

Space-time codes, proposed in [1, 13, 4], were studied initially for TDMA systems, but as they can be applied at symbol-level, we can naturally use them also in WCDMA. Indeed, a two-dimensional space-time block code was adopted for WCDMA in ETSI in 1998 and subsequently in 3G partnership project (3GPP) [18]. More advanced trellis designs, which attempt to maximize the distance of code words, have been recently proposed in [13, 14, 4], holding many of the previously proposed open-loop schemes as a special case. However, the block codes are easy to include in a concatenated coding concept as simple linear processing is required.

In the open-loop concept the channel symbols are divided into two–element blocks and transmitted as $s[2n], s[2n+1]$ and $-s^*[2n+1], s^*[2n]$ from antennas one and two, respectively, at time instants $2n$ and $2n+1$ [1]. These symbols are transmitted using the same spreading code. The received signal in a single-path case can be written as

$$\begin{bmatrix} z[2n] \\ z[2n+1] \end{bmatrix} = \begin{bmatrix} s[2n] & s[2n+1] \\ -s^*[2n+1] & s^*[2n] \end{bmatrix} \begin{bmatrix} h_1 \\ h_2 \end{bmatrix} + \begin{bmatrix} n[2n] \\ n[2n+1] \end{bmatrix} \qquad (1)$$

where it is assumed that the channel remains stationary during the transmission of the successive symbols[14]. The receiver then uses linear orthogonal processing using the estimated channel coefficients $\hat{h}_1$ and $\hat{h}_2$ to detect transmitted symbols as is shown below.

$$\begin{bmatrix} \hat{s}[2n] \\ \hat{s}[2n+1] \end{bmatrix} = \begin{bmatrix} \hat{h}_1 & \hat{h}_2 \\ \hat{h}_2^* & -\hat{h}_1^* \end{bmatrix} \begin{bmatrix} z[2n] \\ z^*[2n+1] \end{bmatrix} \tag{2}$$

In order to maintain full orthogonality the channel estimates need to be perfect, but the loss due to imperfect channel estimates is not severe. In WCDMA systems a rather accurate estimates can be obtained by using the common channel pilots (CPICH) transmitted continuously from the two antennas. This simple symbol level orthogonal coding scheme doubles the diversity of the system without any bandwidth expansion.

2.2 Feedback Modes

Feedback modes are designed to provide additional gain over the open-loop techniques, and in particular over single-antenna transmission, in low mobility environments. These environments may face the most severe capacity bottleneck in the future both due to lack of diversity and due to projected services. Therefore, it can be argued that it is sensible to use a small portion of uplink capacity to achieve significant gains in downlink capacity.

In the feedback modes the UE/terminal feeds back to the network/base station information determining the phase/power settings for the transmit antennas. This information is transmitted in the Feedback Signaling Message (FSM) as a part of the FBI field of uplink dedicated physical control channel (DPCCH). Each message is of length $NW = Npo + Nph$ bits and one bit is transmitted in each slot resulting in 1500 bps signaling overhead.

Mode	1	2
phase bits per word(Npo)	1	3
gain bits per word(Nph)	0	1
feedback bit rate	1500 bps	1500 bps
update rate	1500 Hz	1500 Hz
filtering at BS	yes	no

Table 1: Feedback Mode Parameters

The two feedback modes approximate optimal (e.g. matched) beamformer with a limited feedback channel capacity. Mode 1 maintains equal power transmission for the two antennas by selecting the transmit weight from 4 possible (constant amplitude) values. Mode 2 can provide some additional gain in very low-mobility environments with selection from 16 possible transmit weights (See Fig. 5). Both include novel signaling strategies in order to minimize the control delay in applying the transmit weight. A detailed description of the frame and slot structures can be found in [18].

2.2.1 Preliminaries

It is instructive to first consider the 'optimal' approach in controlling the transmit weights before detailing the particular signaling solutions with low feedback overhead. The unquantized feedback weight can be calculated as follows. Assume for simplicity that there are two transmit antennas each transmitting a common pilot channel (CIPCH). A given terminal obtains channel estimates $\hat{\mathbf{h}}_1$ and $\hat{\mathbf{h}}_2$ for both antennas. Then, the terminal determines how the transmit weights (beam coefficients) in the dedicated channel have to be modified in order to maximize the signal-to-noise ratio. We assume that only one complex coefficient is transmitted to the base station to minimize signaling overhead. It is easy to see that this collapses the the following problem,

$$\mathbf{w} = \arg\max_{\mathbf{w}}(w_1\mathbf{h}_1 + w_2\mathbf{h}_2)^{\mathrm{H}}(w_1\mathbf{h}_1 + w_2\mathbf{h}_2), ||\mathbf{w}||^2 = 1 \qquad (3)$$

Furthermore, as it is only necessary to consider the difference of the channels in order to do optimize the beam, it is sufficient to calculate, *e.g.*, w_2 only, while keeping $w_1 = 1$. One solution is provided by

$$w_2 = ze^{j\phi} = \arg\max_{z,\phi}(\sqrt{1-z^2}\mathbf{h}_1 + ze^{j\phi}\mathbf{h}_2)^{\mathrm{H}}(\sqrt{1-z^2}\mathbf{h}_1 + ze^{j\phi}\mathbf{h}_2),\ z \in [0,1] \qquad (4)$$

which yields

$$\begin{aligned} e^{j\phi} &= \frac{\mathbf{h}_1^{\mathrm{H}}\mathbf{h}_2}{\sqrt{||\mathbf{h}_1^{\mathrm{H}}\mathbf{h}_2||}} \\ z &= \frac{||\mathbf{h}_2||}{\sqrt{||\mathbf{h}_1||^2 + ||\mathbf{h}_2||^2}} \end{aligned} \qquad (5)$$

In the following we describe two approaches on how the FSM message (see Table 1) can be used to signal these weights to the network. Both approaches control only the diversity antenna corresponding to w_2.

2.2.2 Mode 1

In feedback Mode 1 terminal updates the channel estimates, obtained from the common pilot channel (CPICH) every slot. The phase information $\exp(j\hat{\phi})$ is quantized to constellations $\mathcal{S}_1 = \{\exp(j\pi 0), \exp(j\pi)\}$ and $\mathcal{S}_2 = \{\exp(j\pi/2), \exp(-j\pi/2)\}$ at even and odd time slots, respectively. The corresponding feedback commands are '0' and '1', where '0' designates that the first constellation point was closer to phase information. Similarly, '1' designates that the second constellation point was closer to phase information. These are transmitted to the network using the $FSMph$ field. The quantization constellations are known and synchronized for both the terminal and the network. Effectively, the use of two reference constellations $\mathcal{S}_1$ and $\mathcal{S}_2$ results in a concept where a terminal signals the imaginary part and the real part of the downlink channel alternately in successive slots. This concept is similar to set-partitioning, where e.g. the minimum distance between points in multiple constellations is maximized.

This quantization and signaling concept alone enables only crude beamforming with 180 degree resolution. However, the resolution is increased by filtering the feedback signal/weights at the base station. Although different filtering methods are possible the WCDMA concept currently applies simple averaging over two successive weights which enhances the resolution to 90 degrees. Assuming a hard decision is made on each feedback bit prior averaging we end up with four possible weights, i.e., the transmit weight is selected from a QPSK constellation, where transitions only to neighboring weights are possible (due to the memory imposed by the filter).

Since there are only 4 possible weights we can use the dedicated channel pilots (and channel estimate) in order to verify which one of the four weights was in fact transmitted. The verification allows to use the continuous common channel estimate in maximal ratio combining. The actual verification algorithm is not specified in the 3GPP specification[18].

2.2.3 Mode 2

Mode 2 provides more accurate weight signaling to the BS transmitter with 16 possible weights. As opposed to Mode 1, no constellation rotation is done at the terminal and no filtering of the received weights is performed at the base station. The estimated phase difference is quantized to a 8-PSK constellation, and transmitted to the base station using Gray coding to reduce the effects of feedback errors. In addition to adjusting phase, Mode 2 is able to change the relative power of the two antennas using one gain control bit. The weights for antennas 1 and 2 are either $\{0.8, 0.2\}$ or $\{0.2, 0, 8\}$ depending on the value of the $FSMpo$ field, but phase adjustment is applied always to the diversity antenna. The weights are optimized at the terminal on a slot by slot basis. The value of the phase/gain bit per slot is calculated by conditioning on previously transmitted bits. This enables an update rate of once per slot both the terminal and at the base station. Mode 2 relies solely on channel estimation from the dedicated channel as weight verification with 16 states is considered unreliable.

2.2.4 Discussion

Although there are similarities between the two feedback modes it should be noted that they are based on slightly different design philosophies. Mode 1 provides a minimum-delay low resolution channel state information to the transmitter using set-partitioning. The resolution is then enhanced by appropriate filtering. Clearly, the filter length should be matched (among other things) to the channel time-coherence/Doppler for best performance and the Mode 1, with length 2 filter, is a compromise which improves performance over a variety of environments. It is not claimed that the specified filter as such is optimum and further work involves optimizing the FIR (or IIR) at the transmitter. However, Mode 1, as defined above, has only four possible transmit weights and while this may compromise resolution it enables efficient weight verification and subsequently the use of common channel estimates in decoding a given dedicated channel. Mode 2, on the other hand has 16 possible transmit weights and this exacerbates weight verification, and therefore only dedicated pilots are used in channel for a given slot estimation. On the other hand, the weight/beam of the dedicated channel may be better matched to the channel.

Note that both Mode 1 and Mode 2 could be applied analogously in the presence of more that two transmit antennas. To maintain the same overhead in feedback signaling we can apply coherent array transmission where the feedback word designates the relative phase difference between neighboring antenna elements. Furthermore, in environments where the Doppler spread is very low Mode 2 weights could be filtered to further improve resolution. These evolutionary concepts, mostly relevant in indoor environments or pico cells are be evaluated in detail when the structure of common channels are defined in 3GPP standardization for more than two transmit antennas.

3 Soft-weighted space-time transmit diversity

Both Mode 1 and Mode 2 rely heavily on the use of dedicated pilots. Mode 1 requires them for efficient weight verification and Mode 2 for channel estimation and maximal ratio combining in the dedicated channel. As it is possible that some future frame formats do not include dedicated pilots it is interesting to study a feedback-mode which provides robustness to feedback errors by some other means.

In light with 3G acronyms we call the proposed concept Soft-Weighted Space-Time Transmit Diversity (SW-STTD). We describe the soft-weighting below as a form of soft antenna selection but it is clear that e.g. soft beam selection would operate analogously. The received signal in SW-STTD can be written as

$$\begin{bmatrix} z[2n] \\ z[2n+1] \end{bmatrix} = \begin{bmatrix} g_1 s[2n] & g_2 s[2n+1] \\ -g_1 s^*[2n+1] & g_2 s^*[2n] \end{bmatrix} \begin{bmatrix} h_1 \\ h_2 \end{bmatrix} + \begin{bmatrix} n[2n] \\ n[2n+1] \end{bmatrix} \tag{6}$$

which is otherwise identical to STTD except that relative weighting factors g_1 and $g_2 = 1 - g_1$ are imposed on signals transmitted from antennas 1 and 2, respectively. The weight corresponding to the stronger antenna (as measured from CPICH) is set to have the larger weight. This (the result of ranking of the CPICH powers) can be signaled with 1 bit/slot feedback as specified e.g. in [6].

Of course, for optimal reception the receiver needs to know precisely the weights that are applied in the transmitter. This is not possible unless either the feedback channel from the terminal to the base station is perfect, or unless the weight verification procedure (as in Mode 1) is perfect. Rather than devising a weight verification procedure for SW-STTD we propose a simpler solution. We propose that in decoding the SW-STTD transmission the terminal uses STTD processing, thus applying effectively equal gain combining to the two transmit antennas. Furthermore, we propose that the transmit weight is changed based two successive feedback commands, rather than for each command independently. When the base station estimates that feedback bits are [1 1] or [0 0] the transmit gains are set to $[g_1, g_2]'$, where $g_1 > g_2$ or $g_1 < g_2$, respectively. When the successive feedback bits are different, i.e. [1 0] or [0 1], equal power (STTD) transmission is applied, where $g_1 = g_2$. For simplicity we assume that the actual values of the asymmetric weights are fixed *a priori* to $\sqrt{0.8}$ and $\sqrt{0.2}$, which are applied also in Mode 2. Thus, we have effectively defined three possible transmit weights for the STTD encoded signals, $\mathrm{w} = [\sqrt{0.8}, \sqrt{0.2}]'$, $\mathrm{w} = [\sqrt{0.5}, \sqrt{0.5}]'$, and $\mathrm{w} = [\sqrt{0.2}, \sqrt{0.8}]'$. Note that the state transitions reduce the effect of feedback errors since two consecutive errors are needed to create a significant mismatch between the transmit weight and the downlink channel. Clearly, this procedure could be extended to include different state/weight transitions, but we do not pursue this path further in this paper.

4 Simulations

The target of this section is to determine the performance of a transmit diversity system with system imperfections such as channel estimation errors and feedback errors, and therefore we resort to simulations. We compare the performance of open and closed loop modes (including the proposed SW-STTD) and single antenna transmission. Simulation parameters are given in Table 2 and all transmit diversity schemes employ two transmit antennas.

carrier frequency	2150 GHz
chip rate	4.096 Mcps
physical channel rate	32 ksps
data rate	8 ksps
channel estimation	weighted average of pilot symbols over three slots
modulation	QPSK
encoding	rate 1/3, K=9 convolutional
interleaving	10 ms
PCCPCH power	10 times dedicated channel power
PC/FB errors	4%
PC delay	1 slot/ 0.625 ms
FB delay	1–4 slots depending on mode
PC step	1 dB
Pilot symbols/slot	4

Table 2: Simulation Parameters

Bit-error rates of SW-STTD, STTD, Modes 1 and 2, and single antenna transmission are presented in Figures 2 and 3 as a function of transmitted E_s/N_0 with mobile speeds 10 km/h and 42.3 km/h in a single path Rayleigh fading channel and Vehicular A channel, respectively. It is assumed that the two transmit antennas are uncorrelated. With single path channel SW-STTD provides the best performance with 10 km/h and its performance is also close to Mode 1 in the case of 40 km/h mobile speed. However, Mode 1 here assumes ideal Tx weight verification in the mobile station, *i.e.* the mobile always knows the Tx weight even when there are errors in the feedback channel. In practice, the performance of Mode 1 would deteriorate with a fraction of dB (0.2–0.4 dB) due to the imperfect verification.

Average powers of channel taps in the Vehicular A channel model are [0, −2.4, −6.5, −9.4, −12.7] dB. The multipath channel provides diversity to the system so that the transmit diversity methods do not offer as much improvement over single-antenna transmission as in the case of single path channel. Multipath channel and consequently deteriorated channel estimates destroy the symbol level orthogonality of space-time coding so that Modes 1 and 2 clearly outperform SW-STTD and STTD.

5 Conclusion

We have summarized the open and closed loop transmit diversity concepts adopted for the third generation WCDMA system in 3GPP standardization. In addition, we proposed a new concept, soft-weighted space-time transmit diversity (SW-STTD), which improves the system performance over the current WCDMA transmit diversity modes when there is little time diversity offered by multipath propagation. Furthermore, unlike the current WCDMA closed loop transmit diversity modes, SW-STTD does not require dedicated pilot symbols leading to a more efficient usage of the downlink dedicated channels.

References

[1] S. M. Alamouti, "A simple transmit diversity technique for wireless communications," *IEEE Journal on Selected Areas in Communications*, vol. 16, no 8, pp. 1451-1458, October, 1998.

[2] K.Ban, M. Katayama, W. Stark, T. Tamamoto and A. Ogawa, "Convolutionally coded DS/CDMA system using multi-antenna transmission," in *Proceedings of IEEE Globecom '97*, Phoenix, Arizona, Nov. 1997.

[3] W. -L. Kuo and M. P. Fitz, "Design and analysis of transmitter diversity using intentional frequency offset for wireless communications," *IEEE Transactions on Vehicular Technology*, Vol. 46, no. 4, pp. 871-881, November, 1997.

[4] J.-L. Guey, M. P. Fitz, M. R. Bell and Y. Kuo, "Signal design for transmitter diversity wireless communication systems over Rayleigh fading channels," to appear in *IEEE Transactions on Communications.*

[5] A. Hiroike, F. Adachi and N. Nakajima, "Combined effects of phase sweeping transmitter diversity and channel coding," *IEEE Transactions on Vehicular Technology,* Vol. 41, No. 2, pp. 170-176, May 1992.

[6] A. Hottinen and R. Wichman, "Transmit diversity by antenna selection in CDMA downlink," *Proc. IEEE International Symposium of Spread Spectrum Techniques and Applications (ISSSTA)*, Sun City, South Africa, September 1998.

[7] A. Narula, M. Trott and G. Wornell, "Information-theoretic analysis of multiple-antenna transmission diversity," in *Proc. Int. Symp. Inform. Th. Appl.*, Canada, Sept. 1996.

[8] H. Olofsson, M. Almgren and M. Höök, "Transmitter diversity with antenna hopping for wireless communication systems," in *Proceedings of IEEE Vehicular Technology Conference*, Phoenix, Arizona, May 1997.

[9] D. Gerlach and A. Paulraj, "Adaptive transmitting antenna arrays with feedback," *IEEE Signal Processing Letters*, 1: 150-152, October, 1994.

[10] A. J. Paulraj and C. B. Papadias, "Space-time processing for wireless communications," *IEEE Signal Processing Magazine*, pp. 49-83, November, 1997.

[11] G. Raleigh and J. M. Cioffi, "Spatio-temporal coding for wireless communication," *IEEE Transactions on Communications*, Vol. 46, No. 3, March, 1998.

[12] N. Seshadri and J. H. Winters, "Two signaling schemes for improving the error performance of frequency-division-duplex (FDD) transmission systems using transmitter antenna diversity," *International Journal of Wireless Information Networks*, Vol. 1, No. 1, 1994.

[13] V. Tarokh, A. Naguib, N. Seshadri and A. R. Calderbank, "Space-time codes for high data rate wireless communications: performance criteria," in *Proc. IEEE International Conference on Communications*, Montreal, Canada, June 1997.

[14] V. Tarokh, H. Jafarkhani and A. R. Calderbank, "Space-time clock coding for wireless communications: theory of generalized orthogonal designs," *IEEE Transactions on Information Theory*, 1999.

[15] V. Weerackody, "Diversity for the direct-sequence spread spectrum system using multiple transmit antennas," in *Proceedings of IEEE International Conference on Communications*, Geneva, Switzerland, 1993.

[16] J. H. Winters, "Switched diversity with feedback for DPSK mobile radio," *IEEE Transactions on Vehicular Technology,* Vol. 32, pp. 134-150, February 1983.

[17] A. Wittneben, "A new bandwidth efficient transmit antenna modulation diversity scheme for linear digital modulation," in *Proceedings of IEEE International Conference on Communications*, pp. 1630-1634, Geneva, Switzerland, 1993.

[18] 3GPP RAN WG1, Physical Layer- General Desciption, v. 2.0.0, April, 1999.

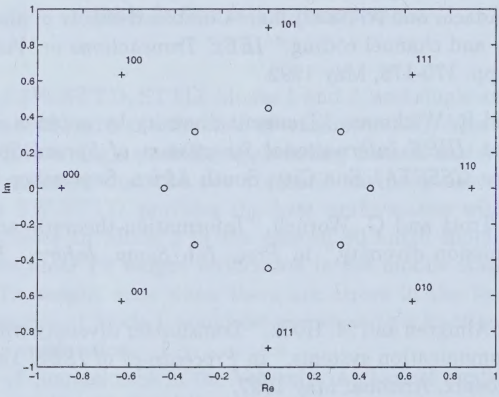


Figure 1: Sixteen possible transmit weights w_2 (with associated feedback bit states) for feedback Mode 2, where the power of each point in the inner constellation is 0.2 and the power of the each point in the outer constellation is 0.8. The resulting transmit weights for the two antennas are thus $\mathrm{w} \in [g\ (1-g)w_2]'$, where $g = \{\sqrt{0.2}, \sqrt{0.8}\}$ and $w_2 \in \{\exp(i\pi\phi/180) | \phi = 180, -135, -90, -45, 0, 45, 90, 135\}$

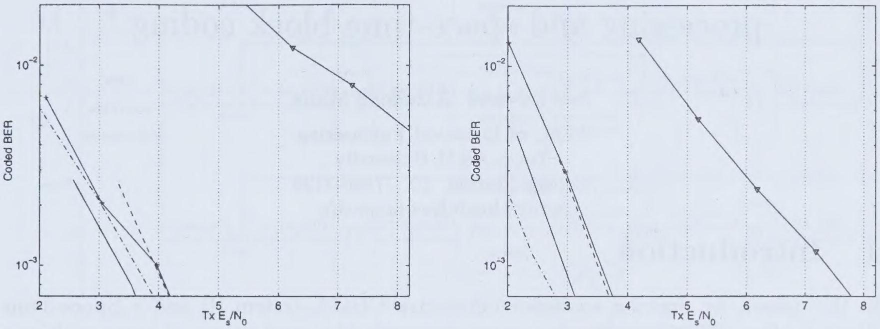


Figure 2: Coded Bit–error rates of SW-STTD (—), STTD (- -), Mode 1 (- · -), Mode 2 (+), and single antenna transmission (∇) as a function of transmitted E_s/N_0 in a flat Rayleigh fading channel with mobile speeds 10 km/h (left) and 40 km/h (right).

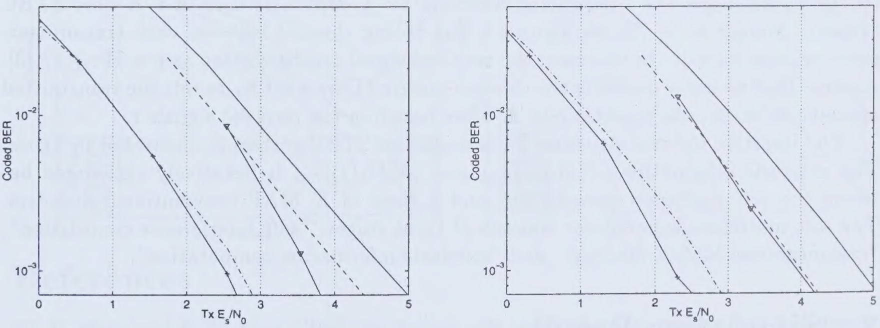


Figure 3: Coded Bit–error rates of SW-STTD (—), STTD (- -), Mode 1 (- · -), Mode 2 (+), and single antenna transmission (∇) as a function of transmitted E_s/N_0 in a Vehicular A channel with mobile speeds 10 km/h (left) and 40 km/h (right).

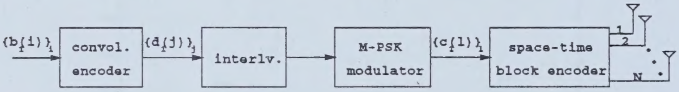
Multiuser detection techniques for combined array processing and space-time block coding *

Ben Lu and Xiaodong Wang
Dept. of Electrical Engineering
Texas A&M University
College Station, TX 77840-3128
wangx,benlu@ee.tamu.edu

1 Introduction

In this paper, by drawing analogies between a CDMA system [1] and a Space-Time Block Coding (STBC) multiuser system, we study the applications of some multiuser detection methods to STBC multiuser systems. Specifically, we show that the so-called "Turbo multiuser detection" technique [1], which performs soft interference cancelation and decoding iteratively, can be applied to STBC multiuser systems in Rayleigh flat-fading channels.

2 Iterative Receiver for Multiuser STBC Systems



Above figure shows the transmitter structure for a particular user in a K-user STBC system. Similar as in [2], we assume a flat fading channel between each transmitter-receiver antenna pair. In this case, the received signal can be written as $\mathbf{r} = \mathbf{H}\mathbf{c} + \mathbf{n}$, [3]. Assume that we know perfectly the channel matrix $\mathbf{H}$, we want to decode the transmitted space-time block code symbols $\mathbf{c}$ of K users based on the received signals $\mathbf{r}$.

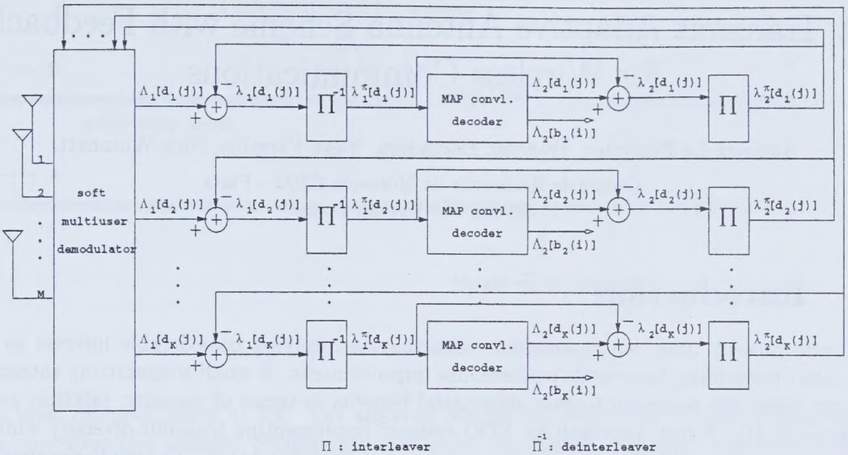
The iterative receiver structure for a multiuser STBC system is illustrated in figure. The extrinsic information $\{\lambda_1^{\pi}[d_k(j)]\}_{j,k}$ and $\{\lambda_2^{\pi}[d_k(j)]\}_{j,k}$ is iteratively exchanged between the soft multiuser demodulator and a bank of K MAP convolutional decoders. The soft multiuser demodulator consists of three stages: "soft interference cancelation", "instantaneous MMSE filtering", and "extrinsic information computation".

3 Simulation Results

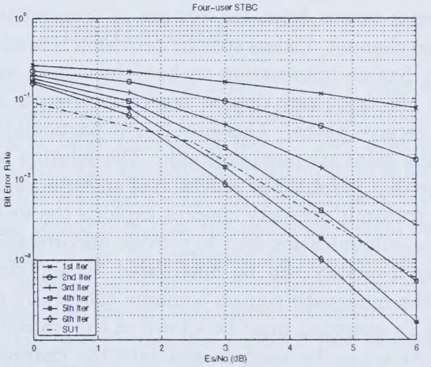
We consider a four users ($K = 4$) STBC system, where each user uses the same (2×2) space-time block code and two transmitter antennas. At the receiver side, there are four receiver antennas. For the purpose of comparison, we also include the performance of the conventional non-iterative receiver (in dash-dot lines and denoted by SU1) in a *single-user* STBC system, which has the same number of transmitter and receiver antennas.

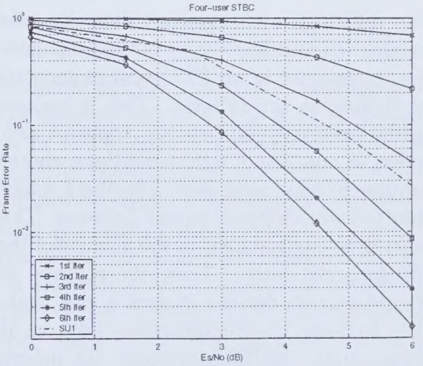
---

*This work was supported in part by the U.S. NSF Grant CAREER CCR–9875314.


1
2
M
soft
multiuser
demodulator
$\Lambda_1[d_1(j)]$
$\lambda_1[d_1(j)]$
Π^{-1}
$\lambda_1^\pi[d_1(j)]$
MAP convl.
decoder
$\Lambda_2[d_1(j)]$
$\lambda_2[d_1(j)]$
$\Lambda_2[b_1(i)]$
Π
$\lambda_2^\pi[d_1(j)]$
$\Lambda_1[d_2(j)]$
$\lambda_1[d_2(j)]$
$\lambda_1^\pi[d_2(j)]$
$\Lambda_2[d_2(j)]$
$\lambda_2[d_2(j)]$
$\Lambda_2[b_2(i)]$
$\lambda_2^\pi[d_2(j)]$
$\Lambda_1[d_K(j)]$
$\lambda_1[d_K(j)]$
$\lambda_1^\pi[d_K(j)]$
$\Lambda_2[d_K(j)]$
$\lambda_2[d_K(j)]$
$\Lambda_2[b_K(i)]$
$\lambda_2^\pi[d_K(j)]$
Π : interleaver
Π^{-1} : deinterleaver

It is seen from simulations that the performance, in terms of both FER and BER, after six iterations is significantly improved compared with that of the non-iterative receiver, (i.e., the performance after the first iteration). More impressively, after iterations the performance of the iterative receiver in a *multiuser* system is even better than that of the conventional receiver in a *single-user* system. It's also shown from our work that the iterative methods can be applied in multiuser space-time trellis coding (STTC) systems and perform very well.





References

[1] X. Wang and H. V. Poor, "Iterative (turbo) soft interference cancelation and decoding for coded CDMA," *IEEE Trans. Commun.*, vol. 47, pp. 1046–1061, July. 1999.

[2] V. Tarokh, A. Naguib, N. Seshadri, and A. R. Calderbank, "Combined array processing and space-time coding," *IEEE Trans. Inform. Theory*, vol. 45, pp. 1121–1128, May 1999.

[3] A. Naguib, N. Seshadri, and A. R. Calderbank, "Applications of space-time block codes and interference suppression for high capacity and high data rate wireless systems," in *Thirty-Second Asilomar Conference on Signals, Systems & Computers*, 1998, pp. 1803–1810.

A Transmit Adaptive Antenna Scheme with Feedback for Wireless Communications

Yannick Le Pézennec, Francesc Boixadera, Yann Farmine, Nick Whinnett.
Centre de Recherche de Motorola CRM - Paris
91193 Gif Sur Yvette, France

1 Introduction

At the present time, smart antenna technology is receiving considerable interest as a means of providing large-scale performance improvements. A small transmitting antenna array offers the potential to gain substantial benefits in terms of capacity, talktime and data rate [1]. A new approach for FDD systems implementing transmit diversity which provides considerable link improvements is presented in this paper. The study developed here aims at demonstrating the feasibility of such systems at frequencies of future wireless systems. Simulations and field tests results will be shown to assess the transmit diversity approach. Note that Motorola CRM has recently introduced similar concepts into the air interface of UMTS/3GPP FDD Downlink.

2 Transmit Diversity Approach

The difficulty with transmit diversity schemes comes from the setting of transmitter weights without knowledge of the propagation channel. Unless specific provision is made, the transmitter in an FDD system generally has no information on how the signals are combining at the receiver. This is because the frequency duplex separation results in fading decorrelation between transmit and receive paths. This problem is particularly severe in non-line of sight conditions where the gains are most desirable. The solution we propose here is based upon a feedback scheme that permits the provision of an estimate of the transmit channels to the Tx array. The transmit diversity concept exploits the local scattering environment in the vicinity of the communication unit to provide uncorrelated transmission paths. Thus, the received signal quality can be maximized by applying weight vectors derived in such a way that the multipath signals combine coherently at Rx. To do so, the channel responses from all Tx antennas must be known at the transmit array point. In TDD systems, Rx information can be used to estimate the channel, however in FDD systems this is not possible. Hence, one possible solution is to use a feedback loop. Considering the case of a TDMA application, the feedback mechanism can be realized as described on figure 1. The reference signals used to probe the Tx antenna channels are mutliplexed in time.

The choice of the combining technique is determined to satisfy the trade-off between performance and cost. We have considered the possibility to apply low complexity sub-optimal approaches which possess some interesting features. *Optimal Weights (OW)* consists in applying an optimal gain and phase adjustment at Tx array which maximizes

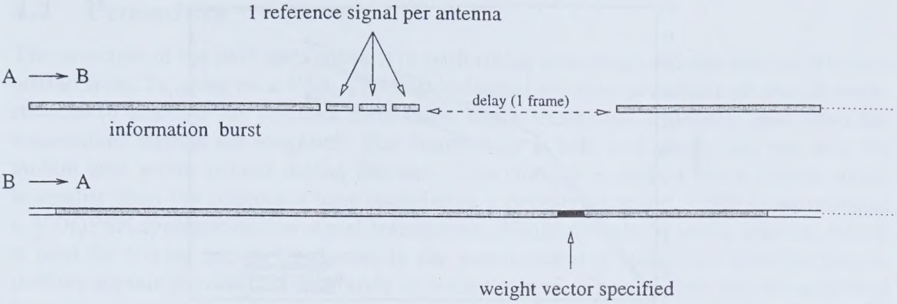


Figure 1: Feedback scheme for an FDD TDMA system.

the total received power at Rx. The antenna weights are set to the conjugate value of the corresponding channel propagation factor for flat fading. *Optimal Selection (OS)* chooses the n out of N best antennas and optimally sets gain and phase. Likewise, *Equal Gain Selection (EGS)* is also a selective technique, but only co-phasing is applied. Lastly, *Selection Diversity (SD)* selects the best antenna channel for transmission of the current burst.

The air interface defined for both simulations and in-the-field measurements can be summarized as follows: an FDD TDMA system operating in narrowband (25 KHz) at 1930 MHz using a $\frac{\pi}{4}$-DQPSK modulation and RRC filters with a roll-off factor of 0.35.

3 Simulations

Simulation results are averaged over a large number of random scenarios. Every scenario is conserved during a time much longer than the channel coherence time so as to average over a large number of fades. The antennas in the transmitting array are assumed isotropic and the antenna configuration is an uniform circular array. Note that no direct path is assumed so that the fading approximates pure Rayleigh. This represents the worst case scenario in that the optimum weight vector will depend most entirely on the fast fading caused by scattering around the antenna array. The simulation results displayed in figure 2 show the performance obtained in flat fading as a function of combining techniques for various numbers of antennas. The Doppler frequency is 5.4 Hz (pedestrian mobile speed) and the feedback link is assumed ideal. Gains are expressed assuming a Bit Error Rate (BER) criterion of 10^{-3}, which represents a reasonable performance objective. In the case of Optimal Selection and Equal Gain Selection methods, the two best antenna channels out of N are selected. The achieved gains are much higher than the beamforming gain and represent the upper bound that can be reached in pure Rayleigh fading conditions when all antenna branches contribute in equally. Note that high gains are attained with a small number of antennas. Likewise, note that sub-optimal approaches perform very well.

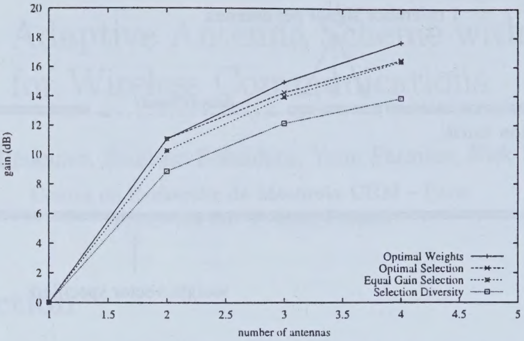


Figure 2: Gain as a function of the number of antennas for various combining techniques, BER criteria of 10^{-3}.

4 Field Tests

4.1 Demonstrator System

Figure 3 displays a diagram of the demonstrator testbed. At Tx site, a computer is used to control the DSP board that generates the digital baseband signals for the four antenna branches. The I-Q signals converted in analog domain are then processed by the RF prototype, which performs the upconversion and mixing. The RF module provides the branch signals in their final form before transmission. No weighting is applied to the various antenna signals because we just handle channel measurements, the weighting process is simulated off-line. At Rx site, the multipath signals combine at a vertical dipole. The resulting signal is amplified by a LNA before being sampled in baseband by a vector signal analyzer (VSA) that acts as a receiver to make time captures of the received signal. The recordings of the antenna channels are processed off-line to evaluate the gain that could be obtained considering an ideal feedback and using appropriate combining schemes.

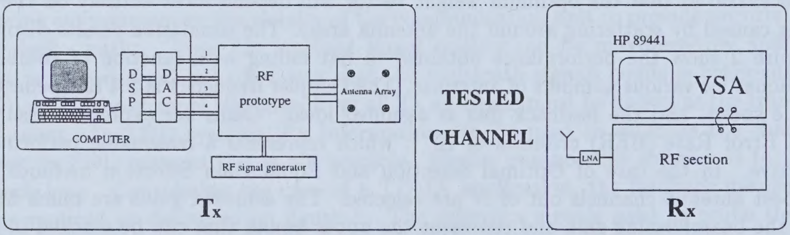


Figure 3: Overall diagram of the demonstrator testbed.

Two different array configurations of four antennas are tested at Tx array. The first array is linear and made of 90 mm long off-the-shelf dipoles vertically polarized with inter-element spacing of 0.2λ. The other antenna configuration is a square array of patch antennas (4.5 x 4.5 cm) circularly polarized with inter-element spacing of 0.2λ.

4.2 Procedure

The principle of the field tests consists in performing recordings of burst sequences transmitted from Tx array on a VSA. The data collected are then processed off-line on workstations to simulate the feedback mechanism and estimate the achievable gain when the transmitted signals are weighted. The transmitter is held in a phone-like way and the mobile user walks around during the test. The channel is probed every 12 ms, which is smaller than the coherence time considering a pedestrian speed. Each burst contains a $\frac{\pi}{4}$-DQPSK synchronization signal transmitted simultaneously by every antenna, which is used for timing recovery purposes in the post-processing stage, and three successive probing signals transmitted separately by each antenna. The diversity gain is calculated from the cummulative distributive function (cdf) of the received power obtained after combining. The resulting cdf curve is compared to the cdf of the best antenna without diversity. Diversity gains are evaluated at $cdf = 0.05$ which means that the same minimum received power level is guaranteed 95% of time.

4.3 Experimental results

Outdoor environments are investigated with the transmitter array at BS. The results are averaged over a large number of different scenarios. No feedback delay is considered between the estimation of the channel and the application of the weighting process. Tests were done in non line-of-sight conditions in order to have Rayleigh fading.

The cdf obtained for the different combining techniques when using the linear array of dipoles is given on figure 4. Figure 5 presents the cdf in the case of the patch array.

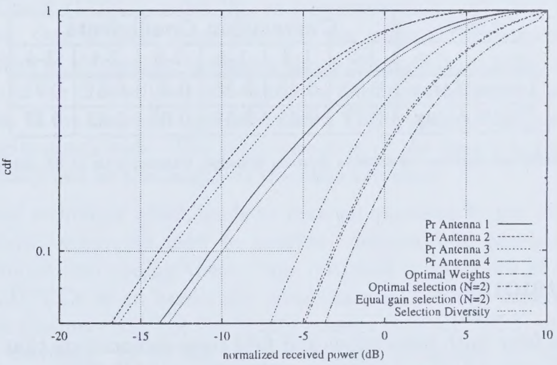


Figure 4: Computed cdf for the linear array of dipole antennas.

The gains achieved with both configurations are summarised in table 1. Results are almost identical in the two cases. An improvement of approximately 10 dB is possible with optimal combining of the branches for a diversity scheme of order 4. The degrees of correlation among the diversity branches are indicated in table 2. The coefficients are relatively high for the patch array. Even if coefficients are smaller for the dipole array, channels remain strongly correlated. This shows that the Transmit Diversity approach permits to realise important gains despite relatively high correlations. This method

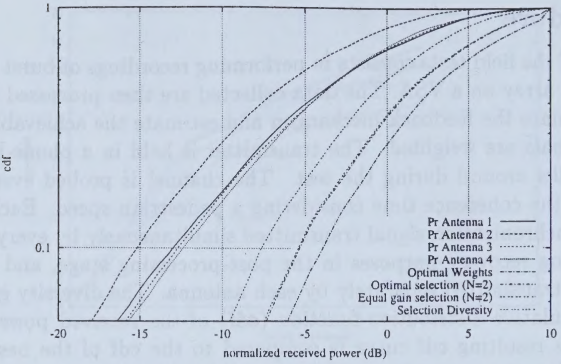


Figure 5: Computed cdf for the square array of patch antennas.

Case	OW	OS	EGS	SD
Linear Array	9.6 dB	8.5 dB	8.4 dB	6.3 dB
Patch Array	9.6 dB	8.3 dB	8.2 dB	6.2 dB

Table 1: Gains measured for a transmit array of four antennas at BS, outdoor conditions, ideal feedback, 5%-tile.

appears to be adequate in order to provide strong link improvements with small array structures.

Case	Correlation Coefficients					
	1-2	1-3	1-4	2-3	2-4	3-4
Linear Array	0.53	0.43	0.35	0.40	0.31	0.41
Patch Array	0.77	0.81	0.52	0.60	0.42	0.77

Table 2: Correlation coefficients between the antenna diversity branches, transmit array at BS, outdoor conditions, NLOS.

5 Conclusions

Promising results from both simulations and field tests demonstrate that transmit antenna diversity is a worthwhile approach to provide substantial link improvements. Selection Diversity and Equal gain Selection appears to be efficient schemes to lower the complexity of such an approach.

References

[1] W.C. Jakes, "Microwave Mobile Communications", *IEEE Press*, 1974.

Concentrate

Tom Richardson and Rüdiger Urbanke
Bell Labs, Lucent Technologies
Murray Hill, NJ 07974

Abstract

A key component of the current asymptotic theory of iterative decoding theorems is the notion of concentration. Concentration is a generalization of the law of large numbers. In the case of randomized code constructions, such as Low-Density Parity-Check codes (LDPCCs) and Turbo codes, concentration states that the performance of the decoding for nearly all codes in the randomized ensemble and all realizations of the noise process in the channel will be close to the average expected performance of the entire ensemble.

In this paper we present an introduction to the relevant techniques used in obtaining such a result and provide a detailed proof for the case of regular LDPCCs. We also point out some of the limitations of the result, indicating directions for future research.

Keywords: turbo codes, message-passing decoders, iterative decoding

1 Introduction

The asymptotic analysis of iterative decoding systems such as Turbo codes [2], and Low-Density Parity-Check (LDPC) codes [5], as presented in [1], [9], and [10] depends on a few basic ideas. One of these ideas concerns the variation among different decoding instances. The class of all possible instances arises not only from different realizations of the noise process in the channel, but also from the freedom residing in the construction of the code used, i.e., in the choice of interleaver for a Turbo-code or the particular graph for a low-density parity-check code. The main assertion of the theory is that this variation decays exponentially fast as the length of the code increases.

The mathematical technique which leads to such an assertion is now virtually standard among probabilistic techniques used to analyse combinatorial problems. The migration of the technique into coding theory was initiated in the work of Luby *et. al.* [1]. They analysed LDPCCs in an essentially combinatorial setting: transmission over the binary symmetric channel (BSC) or the binary erasure channel (BEC) with hard decision message-passing decoding. In [9] the basic technique, as applied to LDPCCs, was significantly generalized to encompass essentially any message-passing decoding algorithm, including belief propagation, and any memoryless channel.

Low-density parity-check codes, like turbo-codes, are codes defined on graphs in which decoding proceeds by passing messages along the edges of the graph. Various generalizations of LDPCCs have been considered: irregular as opposed to regular graphs [1] and codes over non-binary alphabets [6], see also [5]. The ideas presented in this paper apply to all of these generalizations as well as to standard turbo codes and analogous variants of turbo codes [12]. The method of analysis of LDPCCs to which we refer can be summarized as follows:

1. Determine the average behavior of the algorithm under the assumption that the graph determining the code does not contain cycles of small length.
2. Provide a general concentration theorem which guarantees that for almost all codes and almost all inputs the decoder behaves within ϵ of this expected behavior. In particular, if the expected fraction of incorrect messages converges to zero, then this proves that for almost all codes and almost all inputs the decoder will decode correctly.

A further aspect of the analysis, ignored in this paper, concerns the notion of threshold behavior for the decoder in part 1. In this paper we focus on the second part of this general procedure.

The general concentration theorem itself consists of two parts. First one shows convergence of expected performance to the asymptotic value, second one shows concentration of performance as a function of length. This concentration is further separated into concentration over input noise and concentration over the random ensemble of graphs. In practice it is the concentration over the random ensemble of graphs that is of most interest. Ideally one would like to find the graph that offers the best performance averaged over the channel noise statistics. Simulation performance curves convey precisely this information.

The outline of this paper is as follows. In Section 2 we introduce LDPCCs and message-passing decoding. In Section 3 we introduce the Doob martingale process and Azuma's inequality, the two mathematical keys to the concentration theorem, and prove concentration of the chromatic number for random graphs, following Shamir and Spencer [14]. Section 4 contains the proof of concentration for regular LDPCCs. In Section 5 we discuss some of the limitations of the result.

2 Basic Notation and Definitions

In this paper we consider (d_v, d_c)-regular LDPCCs as introduced by Gallager [5]. Let n be the length of the binary code. Construct a bipartite graph with n *variable* nodes on the left and and $m := \frac{nd_v}{d_c}$ *check* nodes on the right. Each variable node represents one bit of the code word and each check node represents one parity check equation. Since each check equation decreases the degrees of freedom by one, it follows that the *design* rate of the code is $r := \frac{n-m}{n} = 1 - d_v/d_c$. The actual rate of a given code may be higher since the check equations might not be independent. There are $nd_v = md_c$ edges in the bipartite graph, d_v edges incident to each variable node on the left and d_c edges incident to each check node on the right. Fig. 1 gives an example of a $(3, 6)$-regular code of length 10. The ensemble $\mathcal{C}_n^{(d_v,d_c)}$ of (d_v, d_c)-regular codes of length n can be generated as follows. Assign to each node d_v or d_c "sockets" according to whether it is a variable node or a check node respectively. Label the variable and check sockets separately with the set $[nd_v] := \{1, \ldots, nd_v\}$ in some arbitrary fashion. Pick a permutation π on nd_v letters at random with uniform probability from the set of all $(nd_v)!$ such permutations. The corresponding (labeled) bipartite graph is then defined by identifying edges as a pairing of sockets and letting the set of such pairs be $\{(i, \pi(i)), i = 1, \ldots, nd_v\}$. This induces a uniform distribution on the set $\mathcal{C}_n^{(d_v,d_c)}$ (the set of labeled bipartite graphs). In practice,

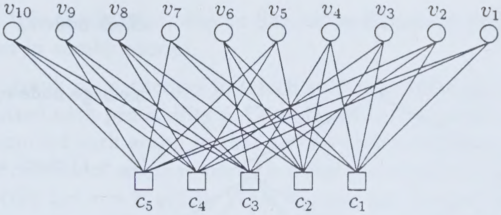


Figure 1: A (3, 6)-regular code of length 10. There are 10 variable nodes and 5 check nodes. The $nd_v = 30 = md_c$ "sockets" of the variable and check nodes are labeled.

one usually modifies the permutation to avoid double edges and excessive overlap of the neighbor sets of the nodes.

Given two nodes in the graph, we say that they have *distance* at most d if they are connected by a sequence of at most d edges. For a given node u, we define its *neighborhood of depth* d, denoted by $\mathcal{N}_u^d$, as the induced subgraph consisting of all nodes at distance at most d from u. Note that for any two nodes u and v we have, by symmetry of the distance function,

$$u \in \mathcal{N}_v^d \iff v \in \mathcal{N}_u^d. \tag{1}$$

Let e be a particular edge between variable node v_i and check node c_j, i.e., $e = (v_i, c_j)$. The *undirected neighborhood of depth* d of e, denoted by $\mathcal{N}_e^d$, is defined as $\mathcal{N}_e^d := \mathcal{N}_{v_i}^d \cup \mathcal{N}_{c_j}^d$. We claim (without proof) that for any pair of edges $e = (v_i, c_j)$ and $e' = (v_{i'}, c_{j'})$,

$$e \in \mathcal{N}_{e'}^d \iff e' \in \mathcal{N}_e^d. \tag{2}$$

The *directed neighborhood of depth* d of $e = (v_i, c_j)$, denoted by $\vec{\mathcal{N}}_e^d$, is defined as the induced subgraph containing the edge e plus the neighborhood of v_i of depth d, with the provision that nodes in this neighborhood are connected to v_i through paths whose first edge is not e. Fig. 2 gives an example of such a directed neighborhood of depth 2. In both cases, if the induced subgraph is a tree then we say that the neighborhood is *tree-like*, otherwise we say that it is not tree-like. Note that the neighborhood is tree-like if and only if all involved nodes are distinct. For $2 \leq d_v < d_c$, one can prove that the number of (distinct) nodes as well as the number of (distinct) edges in a neighborhood of a node or of an edge (undirected or directed) of depth $2l$ is upper bounded by $2(d_v d_c)^l$, see [5, p. 92].

2.1 Symmetry Assumptions and Message Passing

Decoding of LDPCCs consists of rounds of passing messages along the edges of the associated bipartite graph. Each node in the graph receives messages along all of its incident edges and then transmits messages back along each edge. The message-passing paradigm requires that a message sent out along an edge be a function only of messages which have come in along other edges. In the case of variable nodes, outgoing messages may also depend on the received value of the associated bit. The message-passing

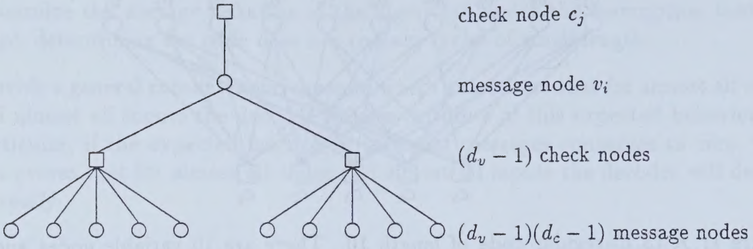


Figure 2: The directed neighborhood of depth 2 of the edge $e = (v_i, c_j)$.

paradigm assumption is critical to the determination of the asymptotic performance of a given ensemble of LDPCCs. It is not critical, however, to the concentration theorem. The concentration theorem also extends to some local algorithms that do not satisfy the paradigm, e.g., flipping algorithms [4].

In the general analysis presented in [9] and [10] certain symmetry assumptions on the channel, the messages, and the update rules are made. These natural assumptions guarantee uniform behavior of the system for all codewords so that it is possible to predict performance of the system from its performance assuming the all-1 codeword was transmitted. These symmetry assumptions are critical to the determination of asymptotic performanced. Although they are not critical to the concentration theorem they lend force to it since concentration of performance on the all-1 codeword then guarantees uniform concentration independent of the codeword.

Since we do not actually require the symmetry assumptions for the sequel we will be brief. Most channels of interest and naturally arising message passing decoding algorithms satisfy certain symmetry properties. We assume antipodal signaling, i.e., the transmitted bit is ±1. Let r denote the received value (e.g., r could the the log-likelihood for the transmitted bit) then we assume channel satisfies the following

$$\textbf{Channel symmetry: } p(r = q|x = 1) = p(r = -q|x = -1).$$

In decoding we assume that the messages m are (possibly restricted) real numbers (again, m could be a log-likelihood for the associated bit.) The symmetry assumptions on the decoder are such that the following holds. If the codeword $b_1, ..., b_n$ were sent, $r_1, ..., r_n$, was received, and $\{m_k\}, k = 1, 2, ...$ are the messages sent in the graph, then the decoding proceeds exactly as if the all-1 codeword was sent, $b_1r_1, ..., b_nr_n$ was received, and each message m_k is replace with b_jm_k where the edge carrying message m_k is associated to bit b_j. (For details see [9].) Under these conditions the (bit or block) error probability is independent of the transmitted codeword. Henceforth, for simplicity, we assume that the symmetry conditions are fulfilled and that the all-1 codeword was transmitted.

3 Doob Martingales and Azuma's Inequality

Although we are most interested in applying the mathematical techniques described in this section to coding, they are most conveniently presented via their application to graph coloring. It was in this application, due to Shamir and Spencer [14], that the method first made its impact in combinatorics.

Consider a random graph on n vertices in which each possible edge, i.e., vertex pair, is independently admitted with probability p. Consider the problem of coloring the vertices of a randomly constructed such graph so that no two vertices connected by an edge have the same color. The number of colors required is called the chromatic number of the graph G and is denoted $\chi(G)$. Let $c = \mathbb{E}[\chi(G)]$. What Shamir and Spencer showed, in effect, is that the chromatic number $\chi(G)$, viewed as a random variable, is tightly concentrated around its expected value c. Perhaps most striking, this result was obtained without the determination of c.

The argument used is now known as an *vertex exposure* martingale argument. Let H denote a graph constructed randomly as above. Pick an arbitrary ordering of the n vertices Imagine that the graph H is unknown but that vertices will be *revealed* one at a time. Each time a vertex is revealed all edges connected to previously revealed vertices are revealed, i.e., when vertex i is revealed the presence or absence of the edge (j, i) for $j < i$ is revealed. Let $X_0(H)$ denote the expected chromatic number of H when nothing has been revealed, i.e., $X_0(H) = c$. Let $X_i(H)$ denote the expected chromatic number of H conditioned on the information obtained after revealing the first i vertices of H. Thus, in particular, $X_n(H) = \chi(H)$. Now, recall that H is a random variable and let X_i denote the random variable obtained from $X_i(H)$.

It is quite intuitive that the sequence $X = X_0, X_1, ..., X_n$ forms a martingale, i.e., $\mathbb{E}[X_j | X_{j-1}, X_{j-2}, ..., X_0] = X_{j-1}$. A martingale constructed this way, as sequence of conditional expectations where the information conditioned on is strictly increasing, is usually called a *Doob's Martingale Process* [8, p. 90]. If we take the graph H and modify the edges involving the ith vertex in an arbitrary way to obtain a graph G, then $|\chi(H) - \chi(G)| \le 1$: given a coloring of H we need at most one more color to color G and vice versa. It follows that $|X_i(H) - X_{i-1}(H)| \le 1$ since $X_{i-1}(H)$ is an average of all possible values for $X_i(H)$, hence, $|X_i - X_{i-1}| \le 1$. Thus, the process X_i behaves like the end position of an unbiased random walk (with memory) with bounded step size.

In the case of a random walk, $X_i = \sum_{j=1}^{i} Z_j$ where the Z_j are independent random variables, we can use the Chernoff bound to give an exponential bound on the probability that X_i will deviate from its mean by more than ϵ. Azuma's inequality provides a similar bound even in the case where the random variables are not independent.

Theorem 1. [Azuma's Inequality] Let $X_0, X_1, \ldots$ be a martingale sequence such that for each $k \ge 1$,

$$|X_k - X_{k-1}| \le \alpha_k,$$

where the constant α_k may depend on k. Then, for all $l \ge 1$ and any $\lambda > 0$

$$\Pr\{|X_l - X_0| \ge \lambda\} \le 2e^{-\frac{\lambda^2}{2\sum_{k=1}^{l} \alpha_k^2}}.$$

Proof. We present the proof for the case $\alpha_k = 1$, following [13].

Set $Y_i = X_i - X_{i-1}$. By convexity of the exponential function we have $e^{\gamma y} \le \cosh(\gamma) + \sinh(\gamma) y$ for all $y \in [-1, 1]$ and any positive γ. First, we have

$$\mathbb{E}[e^{\gamma Y_i} | X_{i-1}] \le \mathbb{E}[\cosh(\gamma) + \sinh(\gamma) Y_i | X_{i-1}] = \cosh(\gamma) \le e^{\gamma^2/2}. \tag{3}$$

We now peel off conditions

$$\mathbb{E}[e^{\gamma(X_m-X_0)}]=\mathbb{E}[\prod_{i=1}^{m}e^{\gamma Y_i}]=\mathbb{E}[(\prod_{i=1}^{m-1}e^{\gamma Y_i})\mathbb{E}(e^{\gamma Y_m}|X_{m-1})]\leq\mathbb{E}[\prod_{i=1}^{m-1}e^{\gamma Y_i}]e^{\gamma^2/2}\leq e^{\gamma^2 m/2}.$$

Finally, setting $\gamma=\lambda/\sqrt{m}$, we have

$$\begin{aligned}\Pr[(X_m-X_0)>\lambda\sqrt{m}] &= \Pr[e^{\gamma(X_m-X_0)}>e^{\gamma\lambda\sqrt{m}}]\\ &\leq \mathbb{E}[e^{\gamma(X_m-X_0)}]e^{-\gamma\lambda\sqrt{m}}\\ &\leq e^{\gamma^2 m/2-\gamma\lambda\sqrt{m}}\\ &= e^{-\lambda^2/2}.\ \square\end{aligned}$$

Returning the the graph coloring problem, the sequence $X_0,\ldots,X_n$ satisfies the conditions required by Azuma's inequality with $\alpha_k=1$. Since $X_0(H)=c$ and $X_n(H)=\chi(H)$, we obtain

$$\Pr[|\chi(H)-c|>\lambda\sqrt{n-1}]\leq 2e^{-\lambda^2/2}.$$

This is Shamir and Spencer's result [14]. For further applications see [8, 13].

4 A General Concentration Theorem

In this section we present the proof of the general concentration theorem for regular LDPCCs. It is important to note that the concentration result applies regardless of whether the average fraction of incorrect messages passed at the l-th decoding step converges to zero or not.

Assume that we are in the l-th decoding step, and recall that the messages passed from message node v_i to check node c_j is a function of the chosen graph and the input to the decoder. Let Z be the number of incorrect messages among all $d_v n$ left-to-right messages sent out in the l-th decoding round. Further, let $\mathbb{E}[Z]$ be the expected value of Z where the expectation is over all graphs and all decoder inputs. For a given edge e whose directed neighborhood is tree-like, let p be the expected number of incorrect messages passed along this edge at the l-th decoding step, averaged over all inputs. In many cases of interest p can be determined [9].

Theorem 2. Over the probability space of all graphs $\mathcal{C}_n^{(d_v,d_c)}$ and channel realizations let Z be the number of incorrect messages and erasures among all nd_v left-to-right messages passed at decoding step l. Then there exists a positive constant β such that for any $\epsilon>0$ and $n>\frac{2\gamma}{\epsilon}$,

$$\Pr\{|Z/(nd_v)-p|>\epsilon\}\leq 2e^{-\beta\epsilon n}. \tag{4}$$

Proof. The proof consists of two parts. The first part shows that $\frac{1}{nd_v}\mathbb{E}[Z]$, the expected average number of incorrect messages, converges to p as n increases. Thus, although we cannot determine $\mathbb{E}[Z]$ for finite n, its asymptotic value can often be determined. The second part of the proof is a concentration argument as outlined in the previous section. We construct a Doob martingale by revealing edges and received values in the graph, prove a bound on the step sizes of the martingale, and apply Azuma's inequality.

Thus, we first write

$$\Pr\{|Z - nd_v p| > nd_v \epsilon\} \leq \Pr\{|Z - \mathbb{E}[Z]| > nd_v \epsilon/2\} + \Pr\{|\mathbb{E}[Z] - nd_v p| > nd_v \epsilon/2\}. \quad (5)$$

The probability that the l-th decoding neighborhood of an edge is not tree-like is at most γ/n for some positive γ (for details see [9]). From this we have $|\mathbb{E}[Z] - nd_v p| \leq d_v \gamma$. It follows that if $n > \frac{2\gamma}{\epsilon}$ then $\Pr\{|\mathbb{E}[Z] - nd_v p| > nd_v \epsilon/2\} = 0$.

We now turn to the concentration argument, i.e., we bound $\Pr\{|Z - \mathbb{E}[Z]| > nd_v \epsilon/2\}$. Recall that, for a particular $(G, R) \in \Omega$, where G is a graph in the ensemble $\mathcal{C}_n^{(d_v,d_c)}$, R is a particular input to the decoder, and Ω is the probability space, Z denotes the number of left-to-right incorrect messages among all nd_v left-to-right messages passed in the l-th decoding step. Let us order the edges by their variable node sockets and reveal them one by one. Following this, let us reveal one by one the received values. Let $=_i$, $0 \leq i \leq (d_v + 1)n = M$, be a sequence of equivalence relations on Ω ordered by refinement, i.e., $(G', R') =_i (G'', R'')$ implies $(G', R') =_{i-1} (G'', R'')$. More specifically, the ith equivalence class represents all graphs and received values for which the first i revealed quantities are identical. Now define $Z_0, Z_1, \ldots, Z_M$ by

$$Z_i(G, R) := \mathbb{E}[Z(G', R')|(G', R') =_i (G, R)].$$

Thus, $Z_i(G, R)$ represents the expected number of incorrect messages conditioned the information revealed in the first i steps, assuming that information is consistent with the particular given G and R. By construction, $Z_0, Z_1, \ldots, Z_M$ is a Doob's martingale process where $Z_0 = \mathbb{E}[Z]$ and $Z_m = Z$. Since we intend to apply Azuma's inequality, we seek to prove a bound

$$|Z_{i+1}(G, R) - Z_i(G, R)| \leq \alpha_i \, , i = 0, \ldots, M - 1, \quad (6)$$

for suitable constants α_i which may depend on d_v, d_c, and l, but preferably not on n.

We first prove (6) for $i \in [nd_v]$, i.e., for the steps where we expose the edges. Let $\mathcal{C}_i$ denote $\bigcup_{k=i+1}^{nd_v} \pi(k)$, i.e., $\mathcal{C}_i$ is the set of check sockets for which no edge has been revealed after the ith step. For $j \in \mathcal{C}_i$, let $\mathcal{G}_j(G, i)$ be the set of graphs in the ensemble $\mathcal{C}_n^{(d_v,d_c)}$ such that the first i edges are equal to the edges in G and $\pi(i + 1) = j$. For each $j, k \in \mathcal{C}_i$ define a map $\phi_{j,k} : \mathcal{G}_j(G, i) \rightarrow \mathcal{G}_k(G, i)$ as follows. For a given graph $H \in \mathcal{G}_j(G, i)$, we construct a graph $\phi_{j,k}(H)$ in $\mathcal{G}_k(G, i)$. The graphs H and $\phi_{j,k}(H)$ are identical except that the endpoints of the $(i + 1)$-th edge and the $\pi^{-1}(k)$-th edge are swapped. More precisely, for H we have $\pi(i + 1) = j$ and $\pi(\pi^{-1}(k)) = k$, whereas if π' denotes the permutation generating $\phi_{j,k}(H)$ we have $\pi'(i + 1) = k$ and $\pi'(\pi^{-1}(k)) = j$. This is shown in Fig. 3. Clearly, $\phi_{j,k}$ is a bijection and, since every (edge labeled) graph in the ensemble has uniform probability, such a bijection preserves probabilities. We claim that

$$|Z(H, R) - Z(\phi_{j,k}(H), R)| \leq 8(d_v d_c)^l \quad (7)$$

for all decoder inputs R. To see this, note that for any edge e the message along e is only a function of the directed neighborhood $\vec{\mathcal{N}}_e^{2l}$. Therefore, a message is only affected by an exchange of the endpoints of two edges if one (or both) of the two edges is (are) in the directed neighborhood $\vec{\mathcal{N}}_e^{2l}$. A neighborhood contains at most $2(d_v d_c)^l$ distinct edges and, by the symmetry property expressed in (2), an edge can be in at most $2(d_v d_c)^l$ neighborhoods. It follows that at most $8(d_v d_c)^l$ neighborhoods can be affected by the exchange of the endpoints of two edges, which proves claim (7).

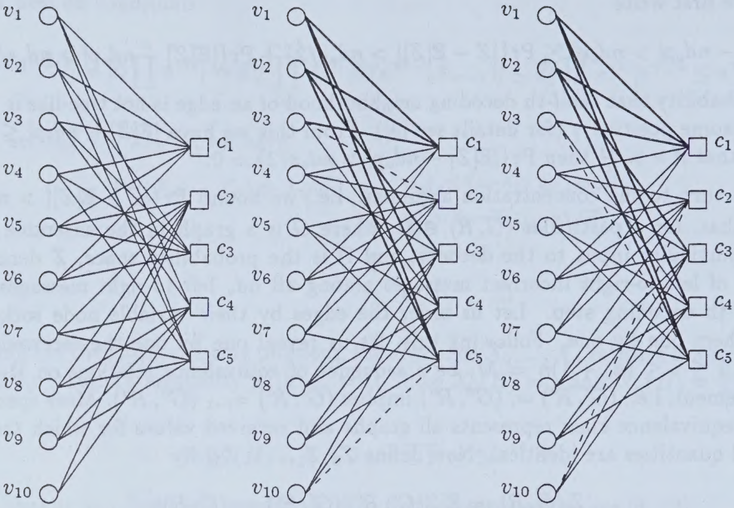


Figure 3: Left: a graph G in the ensemble $\mathcal{C}_{10}^{(3,6)}$; middle: a graph H from the ensemble $\mathcal{G}_2(G,7)$; the first 7 edges that H has in common with G are drawn in bold; right: the associated graph $\phi_{2,5}(H)$. The two dashed lines correspond to the two edges whose end points are switched.

Note that for $i \le nd_v$ we have

$$Z_i(G,R) = \frac{1}{|\mathcal{C}_i|}\sum_{j\in\mathcal{C}_i} \mathbb{E}[Z(G',R')|(G',R') =_i (G,R), \pi(i+1)=j].$$

Since, for some $k \in \mathcal{C}_i$ we have

$$Z_{i+1}(G,R) = \mathbb{E}[Z(G',R')|(G',R') =_i (G,R), \pi(i+1)=k],$$

it follows that

$$\begin{aligned}|Z_{i+1}(G,R) - Z_i(G,R)| &\le \max_{j,k\in\mathcal{C}_i} |\mathbb{E}[Z(G',R')|(G',R') =_i (G,R), \pi(i+1)=j] \\ &-\mathbb{E}[Z(G',R')|(G',R') =_i (G,R), \pi(i+1)=k]| \le 8(d_v d_c)^l,\end{aligned}$$

where, for the last step, we have used (7). This proves (6) for $i \in [nd_v]$ with $\alpha_i = 8(d_v d_c)^l$.

It remains to show that the inequality is also fulfilled for the last n steps. The idea of the proof is very similar. When we reveal a received value at a particular message node v_i then messages whose directed neighborhood include the node v_i can be affected. By the node symmetry property (1) this is equal to the number of nodes in the neighborhood $\mathcal{N}_{v_i}^{2l}$ which is upper bounded by $2(d_v d_c)^l$. This proves (6) for $i \in nd_v+1,\dots,(n+1)d_v$ with $\alpha_i = 2(d_v d_c)^l$.

Theorem 2 now follows by applying Azuma's inequality to Z_i. The parameter $1/\beta$ can be chosen as $68 d_v^{2l+1} d_c^{2l}$. This is by no means the best possible constant. ☐

5 Discussion of the Concentration Theorem

The proof of the concentration theorem is very pessimistic. For example, in the asymptote of the average performance we assume that any loop effectively introduces an error into every message it affects and replacing one received value with another independent one produces an error in every message the replacement affects. In practice, however, this will very rarely be the case, changes tend to disappear under correct decoding. The length required to observe near threshold performance is significantly smaller than any bound that can be obtained from the results presented here.

Similarly, in the martingale argument, we assume that swapping edges leads to an error in every message it effects. Thus, unlike the graph coloring case, the bound used in step (3) is extremely loose. One might ask whether it is possible to significantly improve the results to bring them in line with observed behavior. For example, is it possible to show that the influence of a change in the graph or decays with distance in the graph, i.e., to replace the inequality (3) with something tighter ? Any progress in this direction will necessarily exploit either the statistics of the input distribution or the particular message passing algorithm. The asymptotic performance of Gallager's decoding algorithm A on the BSC and the (3,6) graph, when operating near its threshold, for example, exhibits an instability (see figure 4). A tiny fluctuation in the input error probability can lead to a catastrophic explosion in the number of errors Thus, even in infinitely large graphs, a small perturbation in the input, the flipping of a single bit, can alter the decoding from a correct decoding to an unstable explosion of errors. In such a case, the worst case bounds appearing in the general proof appear to be the best possible.

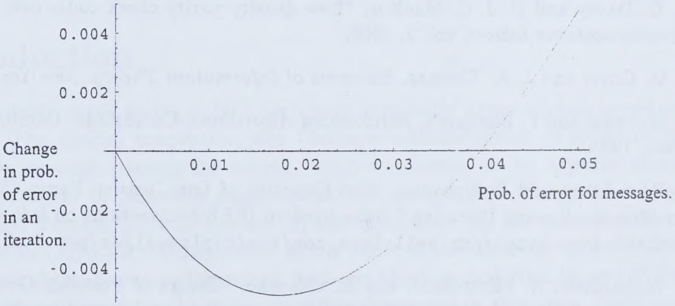


Figure 4: Asymptotic evolution of probability of error using Gallager's decoding algorithm A on a regular (3,6) graph. Input probability of error near 4%.

5.1 Error Floors

Given the known error floors exhibited by turbo-codes and some low-density parity-check codes, it is not surprising that a general theory of such systems must admit the possibility of error floors. The large deviations results provide only bounds on the probability of deviating from the expected value by more than ϵ. It is not possible by this method to show that the probability of error of finite graphs is tending to zero. There other conditions, e.g., expander conditions [4] which can, for the same graph, provide such guarantees. In the case of low-density parity-check codes and turbo-codes error floors can be largely removed or ameliorated by expurgating the ensemble of codes. This must be done with care, however, especially in the case of irregular LDPCCs. Naively constraining the graph can often result in a loss of low SNR performance. This aspect of practical LDPCC design is still something of an art.

References

[1] M. G. Luby, M. Mitzenmacher, M. A. Shokrollahi, and D. A. Spielman, "Analysis of low density codes and improved designs using irregular graphs." available at `http://www.icsi.berkeley.edu/~luby/`.

[2] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon limit error-correcting coding and decoding," in *Proceedings of ICC'93*, (Geneve, Switzerland), pp. 1064–1070, May 1993.

[3] D. J. C. MacKay, "Good error correcting codes based on very sparse matrices," *IEEE Trans. on Information Theory*, vol. 45, pp. 399–431, 1999.

[4] M. Sipser and D. Spielman, "Expander codes," *IEEE Trans. on Information Theory*, vol. 42, November 1996.

[5] R. G. Gallager, *Low-Density Parity-Check Codes.* Cambridge, Massachusetts: M.I.T. Press, 1963.

[6] M. C. Davey and D. J. C. MacKay, "Low density parity check codes over GF(q)," *IEEE Communications Letters*, vol. 2, 1998.

[7] T. M. Cover and J. A. Thomas, *Elements of Information Theory.* New York: Wiley, 1991.

[8] R. Motwani and P. Raghavan, *Randomized Algorithms.* Cambridge: Cambridge University Press, 1995.

[9] T. Richardson and R. Urbanke, "The Capacity of Low-Density Parity-Check Codes under Message-Passing Decoding." Submitted to IEEE transactions on Information Theory. Available from `http://cm.bell-labs.com/who/tjr|ruediger/pub.html`, 1998

[10] T. Richardson, A. Shokrollahi, and R. Urbanke, "Design of Provably Good Low-Density Parity Check Codes." Submitted to IEEE transactions on Information Theory. Available from `http://cm.bell-labs.com/who/tjr|ruediger/pub.html`,

[11] E. A. Gelblum, A. R. Calderbank, and J. Boutros, "Understanding serially concatenated codes from a support tree approach," in *Proceedings of the International Symposium on Turbo Codes and Related Topics,* (Brest, France), pp. 271-274, Sept. 1997.

[12] T. Richardson and R. Urbanke, "The capacity of turbo codes and other concatenated codes under message-passing decoding." in preparation.

[13] N. Alon, J. Spencer, and P. Erdös, *The Probabilistic Method.* New York: John Wiley & Sons, Inc., 1992.

[14] E. Shamir, and J. Spencer, *Sharp concentration of the chromatic number in random graphs* $G_{n,p}$. *Combinatorica* 7, pp. 121-130, 1987.

Efficient Encoding of Low-Density Parity-Check Codes

Tom Richardson and Rüdiger Urbanke
Bell Labs, Lucent Technologies
Murray Hill, NJ 07974

September 21, 1999

Abstract

Low-density parity-check codes can be considered serious competitors to turbo codes in terms of performance and complexity and they are based on a similar philosophy: constrained random code ensembles, and iterative decoding algorithms.

In this paper we consider the encoding problem for low-density parity-check codes. We show how to exploit the sparseness of the parity-check matrix to obtain efficient encoders. In particular we prove that the encoding complexity of all known "good" codes is linear in the block length.

Keywords: turbo codes, parity-check, decoding, encoding.

1 Introduction

Low-density parity-check codes (LDPCCs) were originally invented and investigated by Gallager [1]. The crucial innovation was Gallager's introduction of iterative decoding algorithms (or message passing decoders) which he showed to be capable of achieving a significant fraction of Shannon capacity at low complexity. Except for the papers by Zyablov and Pinsker [2] and Tanner [3] the field then lay dormant for the next 30 years. Interest in LDPCCs was rekindled in the wake of the discovery of turbo-codes and LDPCCs were independently rediscovered by both MacKay and Neal [4] and Wiberg [5].[1] The past few years have brought many new developments in this area. First, in several papers Luby, Mitzenmacher, Shokrollahi, Spielman and Stemann introduced new tools for the investigation of message passing decoders for the binary erasure channel (BEC), [8, 9], and they extended Gallager's definition of LDPCCs to include *irregular* codes, see also [4]. The same authors also exhibited sequences of codes which, asymptotically in the blocklength, provably achieve capacity of a BEC. It was then shown in [10] that similar analytic tools can be used to study the asymptotic behavior of a very broad class of message passing algorithms for a wide class of channels and it was demonstrated in [11] that LDPCCs can come extremely close to capacity for many channels.

In many ways LDPCCs can be considered serious competitors to turbo codes. In particular, LDPCCs exhibit an asymptotically better performance than turbo codes and they admit a wide range of trade-offs between performance and decoding complexity. One major criticism concerning LDPCCs has been their apparent high *encoding* complexity. Whereas turbo codes can be encoded in linear time, a straightforward encoder implementation for a LDPCC has quadratic complexity in the block length. Several authors have addressed this issue:

---

[1] Similar concepts have also appeared in the physics literature [6, 7].

1. It was suggested in [12] to use cascaded rather than bipartite graphs. By choosing the number of stages and the relative size of each stage carefully one can construct codes which are encodable and decodable in linear time. One drawback of this approach lies in the fact that each stage (which acts like a subcode) has a length which is in general considerably smaller than the length of the overall code. This results, in general, in a performance loss compared to a standard LDPCC with the same overall length.
2. In [13] it was suggested to force the parity check matrix to have (almost) upper triangular form, i.e., the ensemble of codes is restricted not only by the degree constraints but also by the constraint that the parity check matrix have upper triangular shape. This restriction guarantees a linear time encoding complexity but, in general, it also results in some loss of performance.

It is the aim of this paper to show that even without cascade constructions or restrictions on the shape of the parity check matrix the encoding complexity is quite manageable in most cases. More precisely, for a $(3,6)$-regular code of length n the encoding complexity seems indeed to be of order n^2 but the actual number of operations required is no more than 0.017^2n^2, and, because of this extremely small constant, even large block lengths result in practically feasible encoders. More importantly, we will show that all known "good" codes have a *linear* encoding complexity and that the required amount of preprocessing is at most of the order $n^{3/2}$.

The proof of these facts is achieved in several stages. We first show in Section 2 that the encoding complexity is upper bounded by g^2, where g, the *gap*, measures in some way to be made precise shortly the "distance" of the given parity check matrix to an upper triangular matrix. In Section 3 we then discuss several greedy algorithms and we show that for these algorithms, when applied to elements of a given ensemble, the gap concentrates around its expected value with high probability. As mentioned above, for the $(3,6)$-regular code the best greedy algorithm which we discuss results in an expected gap of to $0.017n$. Finally, in Section 4 we prove that for all known "good" codes the expected gap is actually of order less than $\sqrt{n}$, resulting in the promised linear encoding complexity.

For a general introduction into low-density parity check codes we refer the reader to [1, 10, 11].

2 Efficient Encoders Based on Approximate Upper Triangulations

In this section we shall develop an algorithm for constructing efficient encoders for LDPC codes. The efficiency of the encoder arises from the sparseness of the parity check matrix H and the algorithm can be applied to any (sparse) H. We note that the algorithm applies generally to matrices H whose entries belong to a field. We assume throughout that the rows of H are linearly independent. In practice this will almost always be the case, and it is a desirable property. If the rows are linearly dependent, then the algorithm which constructs the encoder will detect the dependence and either one can choose a different matrix H or one can eliminate the redundant rows from H in the encoding process.

Let H be a $m \times n$ matrix with $m < n$. If H has the form of an "upper triangular" matrix as shown in Fig. 1(a) then encoding is particularly straightforward: Given a column vector x of length n, where x represents a codeword, choose the last $(n - m)$ components of x in some arbitrary fashion. The remaining m components of x can then be computed by a simple back-substitution algorithm, starting with element x_m. If H is sparse, i.e., the number of non-zero elements is proportional to n, then this procedure results in a linear time encoding algorithm.[2]

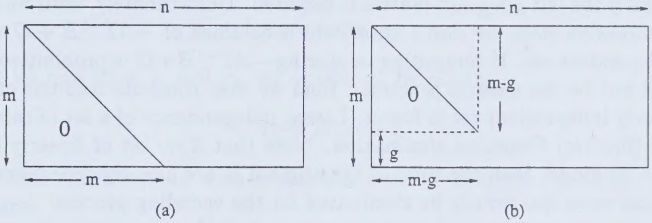


Figure 1: (a) An "upper triangular" parity check matrix H of dimension $m \times n$. (b) A parity check matrix H in "approximate upper triangular form" with gap g. In both cases a top left square portion of the matrix is upper triangular and the entries along the diagonal are non-zero.

Our proposed encoding algorithm is motivated by this ideal scenario. It entails two steps which we will describe in detail in the following sections: A *preprocessing* step which itself is accomplished in two parts and the actual *encoding* step.

2.1 Preprocessing

2.1.1 Phase 1: Approximate Upper Triangulation

The first phase of the preprocessing algorithm is to rearrange the rows and columns of H so that the left $m \times m$ sub-matrix of H is nearly upper triangular as shown in Figure 1(b). More precisely, by reordering the rows and columns of H we bring the parity check matrix into a form such that for some integer g, $0 \le g \le m$, $H_{i,i} \neq 0$ and $H_{i,i-k} = 0$ for $i \in 1, ..., m - g$ and $k \in 1, ..., i - 1$. Hence, H has the form

$$H = \begin{bmatrix} T & B \\ A & C \end{bmatrix},$$

where T is upper triangular and we will say that H is in *approximate upper-triangular* form with *gap* g. Not surprisingly, we will see that g should be made as small as possible.

In Section 3 we will discuss several greedy algorithms and the resulting gaps. For the remainder of this section we assume that an approximate upper-triangulation has been achieved and that the gap is equal to g.

---

[2] Here we assume that n and m are related by $m = (1 - r)n$, where the fixed constant r is the *rate* of the code.

2.1.2 Phase 2: Gaussian Elimination

By using Gaussian elimination we may effectively premultiply H as below.

$$\begin{bmatrix} I & 0 \\ -AT^{-1} & I \end{bmatrix} \begin{bmatrix} T & B \\ A & C \end{bmatrix} = \begin{bmatrix} T & B \\ 0 & -AT^{-1}B + C \end{bmatrix}.$$

(Note that we do not actually compute the matrix AT^{-1} : We use Gaussian elimination to obtain the matrix $-AT^{-1}B + C$ directly.)

By reordering columns, if necessary, we can assume that the left $g \times g$ sub-matrix of $-AT^{-1}B+C$ is non-singular. In practice, the entire sub-matrix $-AT^{-1}B+C$ need not be computed, only the left $g \times g$ sub-matrix is required. Unfortunately, until we have carried out the elimination step, we don't know which columns of $-AT^{-1}B + C$ will form an linearly independent set. If computing or storing $-AT^{-1}B+C$ is prohibitively expensive (this should not be the case in practice), then we may compute columns one at a time until a linearly independent set is found. Linear independence of a set of columns can be checked by (further) Gaussian elimination. Note that if no set of linearly independent columns can be found, then the rows of the original H are linearly dependent. If desired, the redundant rows can simply be eliminated for the encoding process.

Up to a simple reordering of rows and columns of H, we now may assume that H is in the form

$$H = \begin{bmatrix} T & B_1 & B_2 \\ A & C_1 & C_2 \end{bmatrix}, \tag{1}$$

where the left $m \times m$ sub-matrix is

$$\begin{bmatrix} T & B_1 \\ A & C_1 \end{bmatrix}$$

and the $g \times g$ matrix $-AT^{-1}B_1 + C_1$, found by Gaussian elimination, is invertible with inverse Φ, which, in practice, is pre-computed according to the scheme outlined above.

2.2 The Encoding Process

We have constructed all of the objects needed for the encoder and we will now describe how an encoding is actually carried out.

Let $x = (x_a, x_b, x_c)$ be a codeword, i.e., a vector satisfying $Hx = 0$, decomposed according to (1). Thus, x_a has length $m - g$, x_b has length g, and x_c has length $n - m$. Ours will be a systematic code with x_c being the information bits to be encoded. We will now show how to efficiently find x_a and x_b to complete x.

The first step is to compute

$$y_c := B_2 x_c \text{ and } z_c := C_2 x_c\,.$$

This is efficient because B_2 and C_2 are sparse. The second step is to solve

$$-T\hat{x}_a := y_c.$$

for $\hat{x}_a$ using back-substitution. Since T is upper triangular and sparse this is easily done. The third step is to evaluate

$$\hat{s} := A\hat{x}_a + z_c.$$

Since A is sparse this is efficient. Note that $\hat{x}_a = -T^{-1}B_2x_c$ and hence $\hat{s} := (-AT^{-1}B_2 + C_2)x_c$. Now, since x_a and x_b are determined by

$$0 = \begin{bmatrix} T & B_1 & B_2 \\ 0 & -AT^{-1}B_1 + C_1 & -AT^{-1}B_2 + C_2 \end{bmatrix} \begin{bmatrix} x_a \\ x_b \\ x_c \end{bmatrix},$$

we have

$$(-AT^{-1}B_1 + C_1)x_b + (-AT^{-1}B_2 + C_2)x_c = \Phi^{-1}x_b + \hat{s} = 0\,.$$

Thus, the forth step is to set

$$x_b = -\Phi\hat{s}.$$

Finally, since we now have x_c and x_b, we obtain x_a by solving

$$-Tx_a = B_1x_b + y_c$$

for x_a using back substitution. This is efficient since T is upper triangular and sparse and B_1 is sparse.

Roughly speaking, encoding complexity according to the above procedure amounts to performing two multiplications by H, and one by Φ. The critical question is, therefore, how small can we make g? We will address this issue in the next section.

In summary, the pre-computation consists of reordering the rows and columns of H in order to bring H into the form (1) and a Gaussian elimination step which effectively computes the matrix Φ which is the inverse of $-AT^{-1}B_1 + C_1$. For the encoding step we assume that the rows and columns of H are ordered according to (1).

1. Let x_c be the vector of information bits. Set $x_b = 0$ and, using back substitution, compute $\hat{x}_a$ so that the first $m - g$ constraints are satisfied,
2. Let $\hat{s}$ be the vector of parities corresponding to the last g constraints, where $x_a = \hat{x}_a$, $x_b = 0$, and x_c is given.
3. Set $x_b = -\Phi\hat{s}$.
4. Compute x_a by back substitution.

3 Approximate Upper-Triangulation Via Greedy Algorithms

We saw in the previous section that the encoding complexity is of order $n + g^2$, where g is the gap of the approximate upper triangulation. Hence, for a given parity check matrix we are interested in finding an approximate upper triangulation with as small a gap as possible. Given that we are interested in large block-lengths there is little hope of finding the row and column permutation which results in the minimum gap. So we will limit ourselves to *greedy* algorithms.

Another way of stating our problem is to find a large square submatrix of H such that, by permuting columns and rows, this submatrix can be made upper-triangular.

All of the algorithms we shall consider have certain basic steps in common. The algorithms proceed by classifying the constraint nodes C into one of two classes: diagonal constraint nodes C_d and gap constraint nodes C_g. Some variables will be classified in the set of diagonal variables V_d. At all times we will have $|C_d| = |V_d|$ and the submatrix of H induced by C_d and V_d will, modulo permutation of the rows and columns, be upper triangular. After all constraints have been classified, and after suitable reordering, the constraints in C_g comprise the last g rows and the constraints in C_d comprise the first $m-g$ rows as in Fig. 2b and (1). The variables V_d similarly correspond to the first $m-g$ columns.

Each of the algorithms considered has, in part, a recursive structure. As classification proceeds certain portions of the graph are removed. The continuation of the algorithm depends only on the remaining *residual* graph. Typically, the steps the algorithm takes depends on the degree sequence associated to the residual graph. All of the algorithms proceed in the same way when the residual graph has degree one variable nodes. Therefore, we first describe a recursive procedure which operates on graphs having degree one variable nodes.

3.1 Diagonal Extension: Degree One Variable nodes.

Let us assume we have a graph with $\lambda_1 > 0$. (We do not assume here that the edges of the graph are random, although this will be the case for all but one of the algorithms we consider.) In this case we proceed as follows. Let $\mathrm{c}_1, \mathrm{c}_2, ..., \mathrm{c}_k$ be all of the constraints nodes connected to degree one variable nodes and left $\mathrm{v}_1, \mathrm{v}_2, ..., \mathrm{v}_k$ respectively be the, possibly not unique, degree one variable nodes to which they are connected. Let us classify the constraints $\mathrm{c}_1, \mathrm{c}_2, ..., \mathrm{c}_k$ into C_d and the variables $\mathrm{v}_1, \mathrm{v}_2, ..., \mathrm{v}_k$ into V_d. By definition, the submatrix of H consisting of rows associated to $\mathrm{c}_1, \mathrm{c}_2, ..., \mathrm{c}_k$ and columns associated to $\mathrm{v}_1, \mathrm{v}_2, ..., \mathrm{v}_k$, when put in this order, is a diagonal matrix. Furthermore, the columns associated to $\mathrm{v}_i, i = 1, ..., k$ are all 0 except for the entry associated to c_i. Let us reorder the columns of H so that $\mathrm{v}_1, \mathrm{v}_2, ..., \mathrm{v}_k$ are the first k columns and $\mathrm{c}_1, \mathrm{c}_2, ..., \mathrm{c}_k$ are the first k rows. Let H_1 denote the parity check matrix obtained by deleting these k rows and columns from H_1. It is clear that if, by and arbitrary permuting of rows and columns, H_1 can be put in approximate upper triangular form with triangular gap g, then H will have approximate upper triangular form with gap g.

Thus, after classifying the k constraints and variables associated to H, we remove the associated constraint nodes, all of the incident edges, and all variable nodes connected solely to the removed constraints (this includes those associated to $\mathrm{v}_1, ..., \mathrm{v}_k$) from the graph. The remaining *residual* graph is identified with the parity check matrix H_1. If this graph has any degree one variable nodes, then we may iterate the above procedure.

By the *diagonal extension* stage of any algorithm we mean a stage in which the above procedure is iterated until the residual graph is either empty or it has no degree one variable nodes.

3.2 Greedy Algorithm B

Of the six algorithms we investigated we will only present two examples, greedy algorithm B and greedy algorithm D.

Greedy Algorithm B:

1. [Initial Set] Let C_g be initialized to a random subset of the constraints chosen by selecting each constraint independently with probability α (and, hence, C_g initially has size approximately αm). Remove all constraints C_g, the edges emanating from

C_g, and all variable nodes connected solely to C_g from the graph. Let the remaining graph be denoted by G^{init}.

2. [Diagonal Extension] Apply diagonal extension to the parity check matrix induced by G^{init}. Classify any constraint nodes remaining in the final residual graph into C_g.

Lemma 1. [Analysis of Greedy Algorithm B] Let $(\lambda(x), \rho(x))$ be a given degree pair. Given a positive real number x_0, $0 < x_0 < 1$, define the recursion

$$x_l = 1 - \rho(1 - \lambda(x_{l-1}))(1 - x_0). \tag{2}$$

Let x_0^* be the infimum of all positive real numbers x_0 such that the sequence x_l converges to one as l tends to infinity. Pick a code from the ensemble at random and apply greedy algorithm B with $\alpha < x_0^*$. Then (asymptotically in n) the gap is concentrated around the value $(1-r)\alpha n$, hence asymptotically we can achieve the gap $(1-r)x_0^* n$.

There is a strong resemblance of the recursion in (2) with the recursion which determines the threshold of a degree sequence when used over the BEC with a message passing decoder, see [9]. In fact, exchanging the roles of $\lambda(x)$ and $\rho(x)$ and letting $x \to 1 - x$ transforms one recursion into the other. Hence, it is not surprising that the same techniques which were used in [14] to determine exact thresholds can be used here to determine the critical value x_0^*. In many instances x_0^* can be given in terms of radicals or is even a rational number.

Example 1. [Gap for the (3, 6) code and greedy algorithm B] Here, the equation reads

$$x_l = 1 - (1 - x_{l-1}^2)^5 (1 - x_0).$$

In this case the critical value can be determined exactly and is equal to $x_0^* = \frac{22179511}{2^{17}5^5} \sim 0.0541492$. It follows that the gap will be concentrated around $\frac{22179511}{2^{18}5^5} \sim 0.0270746n$. □

Example 2. [A "good" code of maximal degree 12] Assume we want to determine the critical value x_0^* for one of the "good" codes listed in [11]. We pick the code with $\lambda(x) := 0.251\,x + 0.309\,x^2 + 0.001\,x^3 + 0.438\,x^9$ and $\rho(x) := 0.637\,x^6 + 0.363\,x^7$. Quite surprisingly we get $x_0^* = 0$! This means that for any $\epsilon > 0$, we can starting with an $\epsilon/2$ fraction of check nodes and, with exponentially large probability, the final set C_i fulfills $|C_i| \geq (1-\epsilon)n$. Hence, the gap can be made smaller than ϵn for any positive ϵ. We will later proof that it is actually at most the square root the size of the number of variable nodes.

The above example shows that at least for some sequences we have $x_0^* = 0$. When does this happen? A sufficient condition is stated in

Lemma 2. If

$$(1-x) > \rho(1-\lambda(x)),\ \forall x \in (0,1), \tag{3}$$

then $x_0^* = 0$. Further, if equation (3) holds then[3]

$$\rho_2 \lambda'(1) \leq 1 \leq \rho'(1)\lambda_2. \tag{4}$$

---

[3] We note that these condition are also necessary for $x_0^* = 0$.

3.3 Greedy Algorithm D

Algorithm D is similar to algorithm B. The difference is in the way we choose the initial set of constraints.

1. [Initial Set] Selecting at random an α_i fraction of variable nodes with degree i. Take at random $i-1$ of the i neighbors of a selected variable node of degree i and classify them into C_g. All of the constraints classified to C_g, their emanating edges, and variable nodes solely connected to them are removed. Let G^{init} denote the residual graph.

2. [Diagonal Extension] Apply diagonal extension to the parity check matrix induced by G^{init}. Classify any constraint nodes remaining in the final residual graph into C_g.

Lemma 3. [Analysis of Greedy Algorithm D] Let $(\lambda(x), \rho(x))$ be a given degree pair. Let $\tilde{G}$ be the graph obtained from G by taking, for each i, each of the α_i fraction of initially selected variable nodes of degree i and replacing them with i degree one variable nodes, one for each edge. Let $\tilde{\lambda}$ be the resulting left degree sequence. We have, $\tilde{\lambda}_1 = \sum_i \lambda_i \alpha_i$ and, for $i > 1$, $\tilde{\lambda}_i = \lambda_i(1-\alpha_i)$. Define the recursion

$$x_l = 1 - \rho(1 - \tilde{\lambda}(x_{l-1})). \tag{5}$$

Let $q = \sum_i \lambda_i \alpha_i \frac{i-1}{i}$. If x_l converges to one with $x_0 = 0$ then, if we pick a code from the ensemble at random and apply the greedy algorithm D with α as specified, then (asymptotically in n) the gap is concentrated around the value $\sum_j \rho_j/j(1-(1-q)^j)/(\sum_j \rho_j/j)$. Let q^* be the infimum of q achievable under the constraint $1-x > \rho(1-\tilde{\lambda}(x))$ for $x \in (0,1)$. Then, asymptotically, one can achieve the gap $\sum_j \rho_j/j(1-(1-q^*)^j)/(\sum_j \rho_j/j)$.

Example 3. [Gap for the $(3,6)$-code and greedy algorithm D] Let us determine the required size of $\tilde{\lambda}_1$. For $x_l \to 1$ we require, for $x \in (0,1)$,

$$1 - x > (1 - \tilde{\lambda}_1 - (1-\tilde{\lambda}_1)x^2)^5 = (1-\tilde{\lambda}_1)^5(1-x^2)^5$$

which simplifies to

$$1 > (1-\tilde{\lambda}_1)^5(1-x^2)^4(1+x)\,.$$

By differentiating $(1-x^2)^4(1+x)$ we find that it takes its minimum at $x = 1/9$. Thus, the critical value of $\tilde{\lambda}_1$ is $\tilde{\lambda}_1^* = 1 - \frac{81}{80}(\frac{8}{9})^{\frac{1}{5}} = 0.01107$. This corresponds to a gap of $(1-(1-(2/3)*\tilde{\lambda}_1^*)^6) \simeq 0.04348m$ or $0.02174n$.

□

In practice a variant of algorithm D usually gives the smallest gap. This variant can be thought of as a serial version of algorithm D. At each stage, if the residual graph has a degree one variable node then diagonal extension is applied. At stopping sets one selects a variable node of lowest possible degree, i say, from the residual graph, and classifies $i-1$ (assuming no duplicate edges) of its neighbors into C_g. The residual graph now has at least one degree one node and diagonal extension is applied. Applied to the $(3,6)$-code this algorithm gives a gap of $\sim 0.017n$.

4 Good Codes have Linear Encoding Complexity

We have seen in the previous section that degree sequences giving rise to "good" codes will have gaps that are smaller than an arbitrarily small linear fraction of the length of the code. To prove that these codes have linear encoding complexity more work is needed, namely one has to show that the gap g satisfies $g \leq \mathrm{const}\sqrt{m}$ with high probability for large enough m. We will prove

Theorem 1. [Codes with Linear Encoding Complexity] Let (λ, ρ) be a degree sequence pair satisfying $(1-x) > \rho(1-\lambda(x))$ for $x \in (0,1)$, $d_c^{\min} > 2$ and the *strict* inequality $\rho'(1)\lambda_2 > 1$. Let G be chosen at random from the ensemble $\mathcal{C}^m(\lambda, \rho)$. Then, for m sufficiently large, the practical greedy algorithm when applied to G will yield a gap g smaller than $m^{\frac{1}{3}}$ with probability at least $1 - e^{-m^{\frac{1}{4}}}$.[4]

We note that all "good" degree sequence pairs listed in [11] fulfill the above sufficient condition. Further, in experiments when applying the practical greedy algorithm to graphs based on these degree sequences pairs the resulting gap is typically in the range of one to three! This is true even for very large lengths like one million. By correctly choosing the first degree-two variable, the gap can nearly always be lowered to one. The primary reason for these very small gaps is the large number of degree two variable nodes in the good degree sequences. The number of these nodes is sufficiently large so that, with very high probability, the subgraph induced by these nodes has a large (linear size) connected component. Once a single constraint belonging to this component is classified in either G_g or G_d, then the remainder of the component will be classified into G_d during the diagonal extension step. The diagonalization process then typically completes without further increasing the gap.

References

[1] R. G. Gallager, *Low-Density Parity-Check Codes.* Cambridge, Massachusetts: M.I.T. Press, 1963. available at http://justice.mit.edu/people/gallager.html.

[2] V. Zyablov and M. Pinsker, "Estimation of the error-correction complexity of Gallager low-density codes," *Problemy Peredachi Informatsii*, vol. 11, pp. 23–26, Jan. 1975.

[3] R. Tanner, "A recursive approach to low complexity code," *IEEE Trans. Inform. Theory*, pp. 533 – 547, Sept. 1981.

[4] D. J. C. MacKay and R. M. Neal, "Near Shannon limit performance of low density parity check codes," *Electronics Letters*, vol. 32, pp. 1645–1646, August 1996.

[5] N. Wiberg, "Codes and decoding on general graphs." Dissertation No. 440, Dept. Elect. Eng. Linköping University, Sweden, 1996.

[6] N. Sourlas, "?," *Nature*, no. 339, p. 693, 1989.

[7] I. Kanter and D. Saad, "Error-correcting codes that nearly saturate Shannon's bound." 1999.

---

[4] The constants $\frac{1}{3}$ and $\frac{1}{4}$ appearing in the theorem are not the best possible and are chosen largely for convenience.

[8] M. Luby, M. Mitzenmacher, A. Shokrollahi, D. Spielman, and V. Stemann, "Practical loss-resilient codes," in *Proceedings of the 29th annual ACM Symposium on Theory of Computing*, pp. 150–159, 1997.

[9] M. Luby, M. Mitzenmacher, A. Shokrollahi, and D. Spielman, "Analysis of low density codes and improved designs using irregular graphs," in *Proceedings of the 30th Annual ACM Symposium on Theory of Computing*, pp. 249–258, 1998.

[10] T. Richardson and R. Urbanke, "The capacity of low-density parity check codes under message-passing decoding." submitted IEEE IT.

[11] T. Richardson, A. Shokrollahi, and R. Urbanke, "Design of provably good low-density parity check codes." submitted IEEE IT.

[12] D. Spielman, "Linear-time encodeable and decodable error-correcting codes," *IEEE Trans. on Information Theory*, vol. 42, November 1996.

[13] D. J. C. MacKay, S. T. Wilson, and M. C. Davey, "Comparison of constructions of irregular Gallager codes," in *Proceedings of the 36th Allerton Conference on Communication, Control, and Computing, Sept. 1998*, 1998. In press.

[14] T. Richardson and R. Urbanke, "The capacity of turbo codes and other concatenated codes under message-passing decoding." submitted IEEE IT.

Irregular Turbocodes

Brendan J. Frey
Computer Science, University of Waterloo
Electrical and Computer Engineering, University of Illinois at Urbana
http://www.cs.uwaterloo.ca/~frey

David J. C. MacKay
Department of Physics, Cavendish Laboratories
Cambridge University
http://wol.ra.phy.cam.ac.uk/mackay

Abstract

Recently, several groups have increased the coding gain of iteratively decoded Gallager codes (low density parity check codes) by varying the number of parity check equations in which each codeword bit participates. In regular turbocodes, each "systematic bit" participates in exactly 2 trellis sections. We construct irregular turbocodes with systematic bits that participate in varying numbers of trellis sections. These codes can be decoded by the iterative application of the sum-product algorithm (a low-complexity, more general form of the turbodecoding algorithm). By making the original rate 1/2 turbocode of Berrou *et al.* slightly irregular, we obtain a coding gain of 0.15 dB at a block length of $N = 131,072$, bringing the irregular turbocode within 0.3 dB of capacity. Just like regular turbocodes, irregular turbocodes are linear-time encodable.

1 Introduction

Recent work on irregular Gallager codes (low density parity check codes) has shown that by making the codeword bits participate in varying numbers of parity check equations, significant coding gains can be achieved [1–3]. Although Gallager codes have been shown to perform better than turbocodes at BERs below 10^{-5} [4][1], until recently Gallager codes performed over 0.5 dB worse than turbocodes for BERs greater than 10^{-5}. However, in [3], Richardson *et al.* found irregular Gallager codes that perform 0.16 dB *better* than the original turbocode at BERs greater than 10^{-5} [5] for a block length of $N \approx 131,072$.

---

[1] Gallager codes to not exhibit decoding errors, only decoding failures, at long block lengths with $N > 5,000$.

In this paper, we show that by tweaking a turbocode so that it is irregular, we obtain a coding gain of 0.15 dB for a block length of $N = 131,072$. For example, an $N = 131,072$ irregular turbocode achieves $E_b/N_0 = 0.48$ dB at BER $= 10^{-4}$, a performance similar to the best irregular Gallager code published to date [3]. By further optimizing the degree profile, the permuter and the trellis polynomials, we expect to find even better irregular turbocodes. Like their regular cousins, irregular turbocodes exhibit a BER flattening due to low-weight codewords.

2 Irregular turbocodes

In Fig. 1, we show how to view a turbocode so that it can be made irregular. The first picture shows the set of systematic bits (middle row of discs) being fed directly into one convolutional code (the chain at the top) and being permuted before being fed into another convolutional code (the chain at the bottom). For a rate 1/2 turbocode, each constituent convolutional code should be rate 2/3 (which may, for example, be obtained by puncturing a lower-rate convolutional code).

Since the order of the systematic bits is irrelevant, we may also introduce a permuter before the upper convolutional code, as shown in the second picture. In the third picture, we have simply drawn the two permuters and convolutional codes side by side.

For long turbocodes, the values of the initial state and the final state of the convolutional chains do not significantly influence performance (*e.g.*, see [6]). So, as shown in the fourth picture, we can view a turbocode as a code that copies the systematic bits, permutes both sets of these bits, and then feeds them into a convolutional code. We refer to this turbocode as "regular", since each systematic bit is copied exactly once.

The final picture illustrates one way the above turbocode can be made irregular. Some of the systematic bits are "tied" together, in effect causing some systematic bits to be replicated more than once. Notice that too keep the rate of the overall code fixed at 1/2, some extra parity bits must be punctured.

More generally, an *irregular turbocode* has the form shown in Fig. 2, which is a type of "trellis-constrained code" as described in [7]. We specify a *degree profile*, $f_d \in [0, 1], d \in \{1, 2, \ldots, D\}$. f_d is the fraction of codeword bits that have degree d and D is the maximum degree. Each codeword bit with degree d is repeated d times before being fed into the permuter. Several classes of permuter lead to linear-time encodable codes. In particular, if the bits in the convolutional code are partitioned into "systematic bits" and "parity bits", then by connecting each parity bit to a degree 1 codeword bit, we can encode in linear time.

The average codeword bit degree is

$$\bar{d} = \sum_{d=1}^{D} d \cdot f_d \tag{1}$$

The overall rate R of an irregular turbocode is related to the rate R' of the convolutional code and the average degree $\bar{d}$ by

$$\bar{d}(1 - R') = 1 - R. \tag{2}$$

So, if the average degree is increased, the rate of the convolutional code must also be increased to keep the overall rate constant. This can be done by puncturing the convolutional code or by designing a new, higher rate convolutional code.

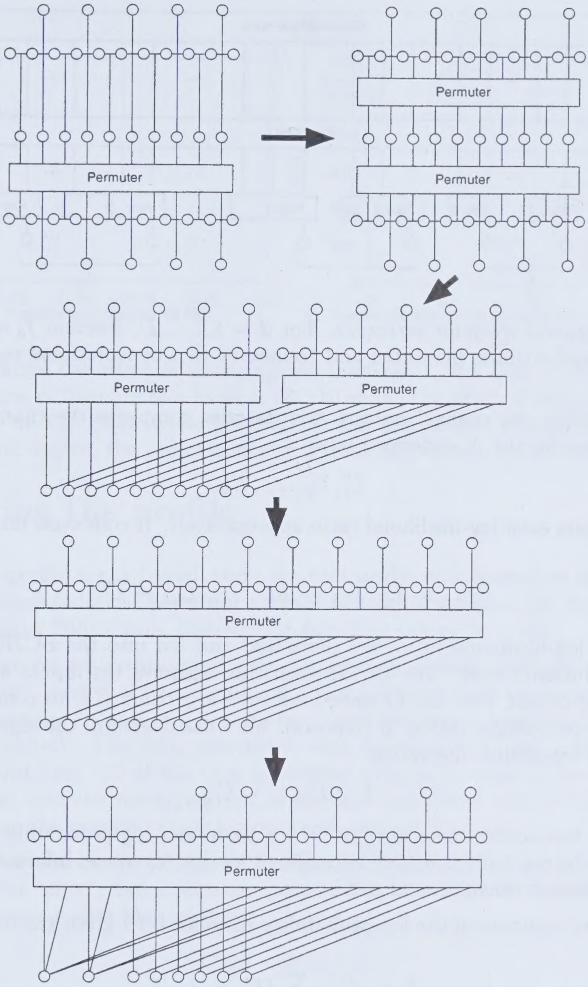


Figure 1: The first 4 pictures show that a turbocode can be viewed as a code that copies the systematic bits, permutes both sets of these bits and then feeds them into a convolutional code. The 5th picture shows how a turbocode can be made irregular by "tying" some of the systematic bits together, *i.e.*, by having some systematic bits replicated more than once. Too keep the rate fixed, some extra parity bits must be punctured. Too keep the block length fixed, we must start with a longer turbocode.

3 Decoding irregular turbocodes

Fig. 2 can be interpreted as the graphical model [6, 8–10] for the irregular turbocode. Decoding consists of the iterative application of the sum-product algorithm (a low-complexity, more general form of turbodecoding) in this graph.

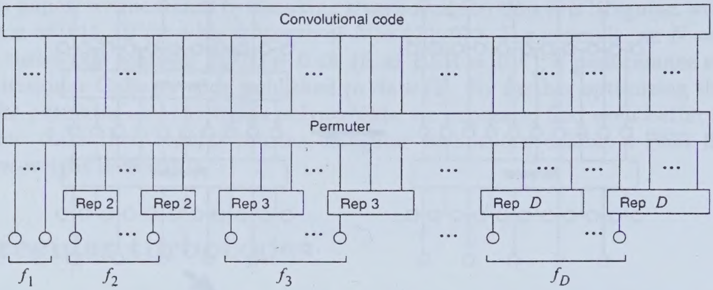


Figure 2: A general *irregular turbocode*. For $d = 1, \dots, D$, fraction f_d of the codeword bits are repeated d times, permuted and connected to a convolutional code.

After receiving the channel output, the decoder computes the channel output log-likelihood ratios for the N codeword bits,

$$L_1^0, L_2^0, \dots, L_N^0, \tag{3}$$

and then repeats each log-likelihood ratio appropriately. If codeword bit i has degree d, we set

$$L_{i,1} \leftarrow L_i^0,\ L_{i,2} \leftarrow L_i^0,\ \dots,\ L_{i,d} \leftarrow L_i^0. \tag{4}$$

Next, the log-likelihood ratios are permuted and fed into the BCJR algorithm [11] for the convolutional code. The BCJR algorithm assumes the inputs are *a priori* log-probability ratios and uses the forward-backward algorithm [12] to compute a set of *a posteriori* log-probability ratios. If codeword bit i has degree d, the algorithm produces d *a posteriori* log-probability ratios,

$$L'_{i,1}, L'_{i,2}, \dots, L'_{i,d}. \tag{5}$$

For a regular turbocode, there are just two *a posteriori* log-probability ratios, $L'_{i,1}$ and $L'_{i,2}$, for each degree 2 bit and they correspond to the "extrinsic information" produced by each constituent convolutional code.

The current estimate of the log-probability ratio for bit i given the channel output is

$$\hat{L}_i \leftarrow L_i^0 + \sum_{k=1}^{d} (L'_{i,k} - L_{i,k}). \tag{6}$$

To compute the inputs to the BCJR algorithm needed for the next iteration, we subtract off the corresponding outputs from the BCJR algorithm produced by the previous iteration:

$$L_{i,k} \leftarrow \hat{L}_i - L'_{i,k}. \tag{7}$$

So, each iteration consists of computing the inputs to the BCJR algorithm, permuting the inputs, applying the BCJR algorithm, permuting the outputs of the BCJR algorithm, and taking the repetitions into account to combine the outputs to form estimates of the log-probability ratios of the codeword bits given the channel output.

In our simulations, after each iteration, we check to see if the current decision gives a codeword. If it does, the iterations terminate and otherwise, the decoder iterates further until some maximum number of iterations is reached and a decoding failure is declared.

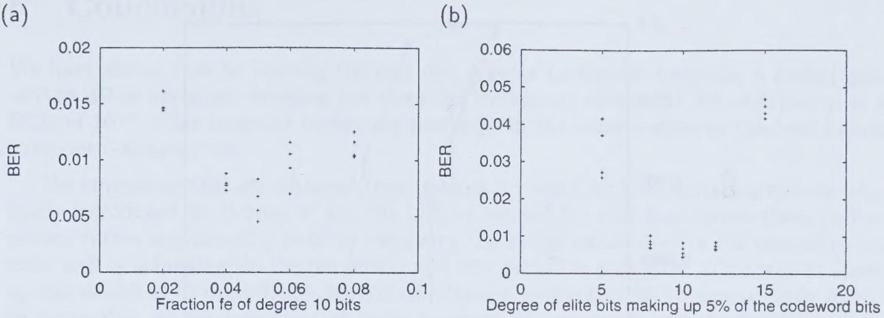


Figure 3: (a) shows the effect of changing the fraction of elite bits on the BER, while keeping the degree of the elite bits fixed at 10. (b) shows the effect of changing the degree of the elite bits on the BER, while keeping the fraction of elite bits fixed at 0.05. For each fraction and degree, the performance of 4 randomly drawn permuters is shown.

4 Selecting the profile

Finding a good profile is not trivial, since the best profile will depend on the parameters of the convolutional code, the permuter and the distortion measure (bit error rate, block error rate, decoding failure rate, high-weight decoding failure rate, *etc.*)

The results we report in this paper were obtained by making small changes to a block length $N = 10,000$ version of the original rate $R = 1/2$ turbocode proposed by Berrou *et al.*. In this turbocode, $f_1 = f_2 = 1/2$ (see Fig. 2) and the convolutional code polynomials are 37 and 21 (octal). The taps associated with polynomial 37 are connected to the degree 2 codeword bits, 1/2 of the taps associated with polynomial 21 are connected to the degree 1 bits, and the remaining 1/2 of the taps associated with polynomial 21 are punctured, giving the required convolutional code rate of $R' = 2/3$.

To simplify our search, we considered profiles where besides degrees 1 and 2, only one other degree, e for "elite", had a nonzero fraction. So, for a code with overall rate R and fraction f_e of degree e elite bits, we have

$$\begin{aligned} f_1 &= 1 - R = 1/2, \\ f_2 &= 1 - f_1 - f_e = 1/2 - f_e. \end{aligned} \tag{8}$$

In this restricted class of codes, irregularity is governed by two parameters, e and f_e.

From (2) it is clear that when the average degree is increased, the rate of the convolutional code must also be increased to keep the overall rate at 1/2. We increased the rate of the punctured convolutional code by further puncturing the taps associated with polynomial 21 to obtain a convolutional code with rate

$$R' = 1 - \frac{1-R}{\bar{d}} = 1 - \frac{1/2}{1/2 + 2(1/2 - f_e) + e f_e}. \tag{9}$$

So, in the codes we explored, the level of puncturing was quite high and some extra low-weight codewords were introduced.

To begin with, we made an irregular turbocode with $e = 10$ (chosen using intuition) and varied f_e from 0.02 to 0.08 while measuring the BER at $E_b/N_0 = 0.6$ dB. In each

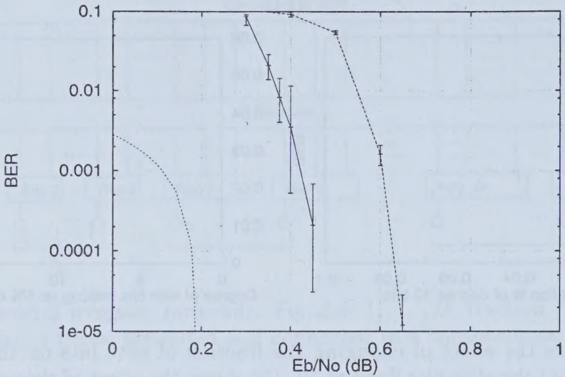


Figure 4: Performance of the original block length $N = 131,072$ turbocode (dashed line) and one of its irregular cousins (solid line). These results are for irregular turbocodes obtained by tweaking the original turbocode – we are currently searching for optimal degree profiles, permuters and trellis polynomials.

experiment, we simulated enough blocks to obtain a relatively small confidence interval. The results are shown in Fig. 3a, which indicates that for degree 10 elite bits, the best fraction is roughly 0.05. Next, we kept the fraction of elite bits fixed at $f_e = 0.05$ and we varied the degree of the elite bits. The results are shown in Fig. 3b, which indicates that for a fraction of 0.05, the best degree is roughly 10.

These results show that for $e = 10$, $f_e = 0.05$ is a good fraction and that for $f_e = 0.05$, $e = 10$ is a good degree. However, values of e and f_e that give good profiles are probably correlated, so we are currently extending our search.

5 Results

Fig. 4 shows the simulated BER-E_b/N_0 curves for the original block length $N = 131,072$ regular turbocode (dashed line) and its irregular cousin (solid line), using profile $e = 10$, $f_e = 0.05$.

The irregular turbocode clearly performs better than the regular turbocode for BER $> 10^{-4}$. At BER $= 10^{-4}$, the $N = 131,072$ irregular turbocode is 0.3 dB from capacity, a 0.15 dB improvement over the regular turbocode.

For high E_b/N_0, most of the errors for the irregular turbocode were due to low-weight codewords. According to preliminary results, the distribution of error weights appears to indicate that the flattening effect for the particular $N = 131,072$ irregular turbocode we constructed occurs at a *higher* BER than it does for the regular turbocode. However, the flattening effect is highly sensitive to the technique used to construct the permuter (we drew it at random) and the design of the convolutional code (we just further punctured the convolutional code used in the original turbocode). We are currently experimenting with techniques for lowering the level of the flattening effect (*e.g.*, see [13]).

6 Conclusions

We have shown that by making the original, regular turbocode irregular, a coding gain of 0.15 dB is obtained, bringing the irregular turbocode within 0.3 dB of capacity at a BER of 10^{-4}. This irregular turbocode performs in the same regime as the best known irregular Gallager code.

We emphasize that we obtained these results by tweaking the regular turbocode originally introduced by Berrou *et al.*. We believe we will be able to improve these performance curves significantly, both by exploring the polynomials used in the convolutional code and by adjusting the degree profile and the permuter structure. (One way to speed up the search is to extend the method of "density evolution" [3] to models with state.) In particular, we are investigating ways to select the permuter and the polynomials to eliminate low-weight codewords, thus reducing the flattening effect [13–15]. We are also studying ways of constraining the degree 1 "parity" bits (*i.e.*, increasing their degree) to eliminate low-weight codewords.

References

[1] M. G. Luby, M. Mitzenmacher, M. A. Shokrollahi, and D. A. Spielman, "Improved low-density parity-check codes using irregular graphs and belief propagation," in *Proceedings of IEEE International Symposium on Information Theory*, 1998.

[2] D. J. C. MacKay, S. T. Wilson, and M. C. Davey, "Comparison of constructions of irregular Gallager codes," *IEEE Transactions on Communications*, vol. 47, October 1999.

[3] T. Richardson, A. Shokrollahi, and R. Urbanke, "Design of provably good low density parity check codes." Submitted to *IEEE Transactions on Information Theory*, July 1999.

[4] D. J. C. MacKay, "Good error-correcting codes based on very sparse matrices," *IEEE Transactions on Information Theory*, vol. 45, pp. 399–431, March 1999.

[5] C. Berrou and A. Glavieux, "Near optimum error correcting coding and decoding: Turbo-codes," *IEEE Transactions on Communications*, vol. 44, pp. 1261–1271, October 1996.

[6] B. J. Frey, *Graphical Models for Machine Learning and Digital Communication*. Cambridge MA.: MIT Press, 1998. See `http://www.cs.utoronto.ca/~frey`.

[7] B. J. Frey and D. J. C. MacKay, "Trellis-constrained codes," in *Proceedings of the* 35^{th} *Allerton Conference on Communication, Control and Computing 1997*, 1998. Available at `http://www.cs.utoronto.ca/~frey`.

[8] N. Wiberg, H.-A. Loeliger, and R. Kötter, "Codes and iterative decoding on general graphs," *European Transactions on Telecommunications*, vol. 6, pp. 513–525, September/October 1995.

[9] F. R. Kschischang and B. J. Frey, "Iterative decoding of compound codes by probability propagation in graphical models," *IEEE Journal on Selected Areas in Communications*, vol. 16, pp. 219–230, February 1998.

[10] R. J. McEliece, D. J. C. MacKay, and J. F. Cheng, "Turbo-decoding as an instance of Pearl's 'belief propagation' algorithm," *IEEE Journal on Selected Areas in Communications*, vol. 16, February 1998.

[11] L. R. Bahl, J. Cocke, F. Jelinek, and J. Raviv, "Optimal decoding of linear codes for minimizing symbol error rate," *IEEE Transactions on Information Theory*, vol. 20, pp. 284–287, March 1974.

[12] L. E. Baum and T. Petrie, "Statistical inference for probabilistic functions of finite state markov chains," *Annals of Mathematical Statistics*, vol. 37, pp. 1559–1563, 1966.

[13] M. Öberg and P. H. Siegel, "Application of distance spectrum analysis to turbocode performance improvement," in *Proceedings of the* 35th *Allerton Conference on Communication, Control and Computing 1997*, 1998.

[14] D. Divsalar and F. Pollara, "Turbo-codes for PCS applications," in *Proceedings of the International Conference on Communications*, pp. 54–59, 1995.

[15] D. Divsalar and R. J. McEliece, "Effective free distance of turbocodes," *Electronics Letters*, vol. 32, pp. 445–446, February 1996.

On Quasi-Cyclic Repeat-Accumulate Codes

R. Michael Tanner
Department of Computer Science
University of California, Santa Cruz
Santa Cruz, CA 95064
tanner@cse.ucsc.edu

Abstract

Repeat-Accumulate (RA) codes, consisting of a simple binary repetition followed by a permutation and an accumulator, are converting into quasi-cyclic codes with multiple cross-linked cycles. The algebraic heuristics are used to design codes with an explicit algebraic permutation yielding desirable graph properties. Recursive graph constructions techniques are given for the construction of very long codes. Simulation results for codes of rate 1/3 and 1/4 demonstrate that these algebraic RA codes can compete with RA codes using pseudo-random permutations and outperform the random codes at high SNR.

1 Introduction

Divsalar, Jin, and McEliece [3] define a class of "turbo-like" codes [2] called *repeat and accumulate* (RA) codes in which an input block of N bits is repeated q times, then passed through a permutation or scrambler, and then input to a rate 1 two-state convolutional encoder, a simple accumulator. In some sense, RA codes are the simplest possible serial concatenated code of the turbo code style: neither the repetition nor the trivial convolutional code by itself achieves any coding gain. Nonetheless, the authors demonstrate that for sufficiently long permutations, RA codes perform remarkably well. By calculating the explicit input-output weight enumerator for these codes, they are able to show that for the ensemble of codes with randomly chosen permutations, the probability of block error for maximum likelihood decoding approaches zero as the block length increases to infinity at sufficiently high E_b/N_0. As the repetition factor q increases and the code rate decreases, the gap between their bounds and the Shannon limit decreases monotonically. The theoretical results for RA codes are supported by empirical studies showing that suboptimal iterative decoders also perform surprisingly well. Indeed, for code rates of $1/q, q \geq 3$, RA codes are an interesting option for a very low complexity system.

For a given rate and block length, an RA code system is defined entirely by the permutation employed. While a random permutation on average is good, any specific permutation may engender some number of relatively low weight codewords. The presence of low weight words precludes very low output bit error rates (BER) at high signal-to-noise ratios, since the low weight words dominate the sum in union bounds on the bit error rate [1]. Random permutations are also problematic due to the memory required to store the addresses to which each input bit is mapped. It would be preferable to have a compact algorithm to generate the addresses, and to be able to prove that the block code's minimum distance is large enough to not pose a significant source of error at higher E_b/N_0.

In this paper we give an algebraic prescription for a rate $1/q$ RA code of block length $N = qK$ when $qK = p(p-1)$ for some prime p such that q divides $p-1$. These codes do have a compact description and are amenable to algebraic analysis of the code space. We have not yet been able to devise a general bound on minimum distance, but simulations studies of two examples indicate that the minimum distance is superior to that of random codes. The program is as follows: A length N RA code is fashioned into a quasi-cyclic repeat-accumulate code (QCRA): first by tailbiting, requiring that the starting state and the ending state of the accumulator be the same, not necessarily zero; second, by imposing a regular periodicity on the permutation connections. The algebraically determined periodicity gives the code constraint (Tanner) graph a useful automorphism group. Effectively, the QCRA code is an algebraic time-varying tailbiting convolutional code, or equivalently, a quasi-cyclic code, with a very restricted form. To define long block length QCRA codes, a constraint graph is described recursively starting from a small graph with suitable valencies using what we will call a *quasi-cyclic extension*, or an *n-cyclic extension*, to make evident the length of the shift invariant cycles. This procedure yields a quasi-cyclic graph whose adjacency matrix is amenable to Fourier transform analysis. The recursive construction gives insights into the problem of constructing a code graph with comparatively few short cycle, assuring that the initial iterations of an iterative decoder will yield statistically independent estimates for a decoded bit. For some short codes, it is possible to prove a bound on minimum distance algebraically or computationally, exploiting the automorphism group. The codes of our prescription must have a minimum distance less than $q\sqrt{N}$. Simulations results for two algebraic QCRA codes show that they perform better than two random permutation competitors at high E_b/N_0, indicating that the minimum distance is almost certainly larger.

2 RA codes, the accumulator chain, and low weight words

Figure 1 gives a circuit diagram for a rate $1/q$ binary RA code and the associated code constraint graph. The information bits are of degree q (shown with q=3), the repetition factor. Codeword bits sent to the channel are shown as bit nodes of degree 2. While in a systematic version the information bits themselves are sent, we will assume that only the channel bits are transmitted. The simplicity of the rate 1 accumulator code is manifest in the constraint graph. The recursive equation of the accumulator imposes a parity check equation involving three variables: the current output channel bit value, the next input information bit value, and the next output channel bit value. An RA code constraint graph therefore always consists of a sequence of channel bits interspersed in alternation between parity nodes of degree 3, with the exact connection of the degree q information bit nodes to the parity nodes determined by the permutation used. The connection to a node will be referred to as a "tap" in the accumulator chain. Each information bit has q taps into the accumulator chain. After all the permuted information bits have entered the accumulator, it may not be in the zero state, and there is a practical question of how to terminate the transmission. One strategy is simply to allow some finite tail, repeating the final output value, and then to stop. This can induce minor end effects, where information bits with taps near the end of the transmission are more or less susceptible to error.

Arbitrarily viewing the time axis as moving from left to right, when the accumulator

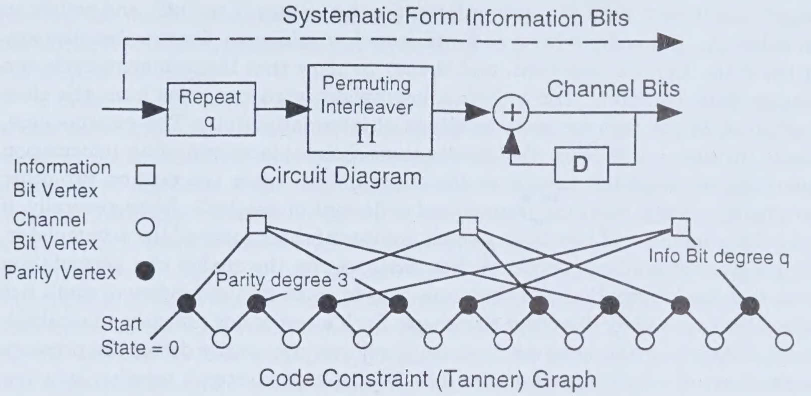
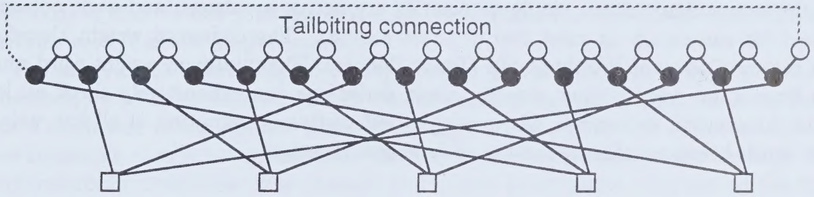

Systematic Form Information Bits
Repeat
q
Permuting
Interleaver
Π
Channel Bits
+
D
Circuit Diagram
Information
Bit Vertex
Channel
Bit Vertex
Parity Vertex
Parity degree 3
Info Bit degree q
Start
State = 0
Code Constraint (Tanner) Graph

Fig. 1 Repeat-Accumulate Code Circuit and Graph


Tailbiting connection

$$H = \begin{array}{c} \\ \end{array}\begin{array}{cccc} \text{Info.} & \text{Channel} & & \\ \end{array}$$

$$H = \begin{vmatrix} I_0 & I_0 & 0 & I_1 \\ I_3 & I_0 & I_0 & 0 \\ I_2 & 0 & I_0 & I_0 \end{vmatrix}$$

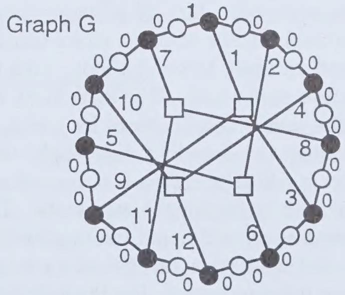
Parity-Check Matrix
5x5 Circulant Blocks

$$I_s = \;\downarrow s\; \begin{vmatrix} 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{vmatrix} \qquad s=2$$

Fig. 2 [15,5,7] Algebraic Quasi-Cylic RA Code

$$A = \begin{vmatrix} 1 & 2 & 4 & 8 \\ 3 & 6 & 12 & 11 \\ 9 & 5 & 10 & 7 \end{vmatrix}$$

Fig. 3 ***13*-cyclic extension**
N=12(13), q=3


Graph G
1
7
1
2
10
4
5
8
9
3
11
6
12
0

is started in the zero state, the output channel bits will remain zero until the first non-zero bit causes the accumulator output to turn "on" and start emitting ones. It will continue to output ones until the next information bit causes it to "off' and return to the zero state. An RA codes with $q = 2$ will have low minimum distance because any cycle in the graph forms a codeword, and is easy to show that the minimum cycle can be no longer than $O(logN)$. For $q \geq 3$ a low weight word can arise from the close juxtaposition of all the taps for some small set of information bits. The extreme case, for example, with $q = 4$, is when the permutation chosen places two of an information bit's connection on successive parities in the chain and the other two taps on two other successive parity checks, resulting in a transmitted codeword of weight 2. More generally, if all the taps for a small set of bits lie in a small number of short zones of the accumulator, a low weight channel word can result. A first order test for the quality of a permutation is to check that the information bits that have taps in some sliding window of small size have their other taps widely dispersed elsewhere. Such a test is not sufficient to establish a minimum distance bound, however, because it will not necessarily detect the presence of some small set of non-zero information bits having taps clustered together in a few narrow zones.

A computational search for the minimum distance of an RA code can take advantage of a simple graph argument. For t non-zero information bits, there are qt ones entered into the accumulator chain, imply qt parities are active equations, and each non-zero channel bit can satisfy at most two of these parities. The codeword weight therefore must satisfy *Codeword Weight* $\geq q/2$ *(Input Weight).* The minimum weight word must arise from a low weight input, and for short codes one can exhaustively check all low weight information sequences. Obviously, an exhaustive examination of all low weight input words becomes infeasible as the code length increases.

3 RA Codes in Quasi-Cyclic Form

A cyclic code of length N is one for which the code automorphism group contains a cyclic group of order N that is one-transitive on the bits, allowing the bits to be organized into a single cycle of length N. Any cyclic shift $mod\ N$ preserves the codespace. A *quasi-cyclic code* of length N is one for which the code is preserved by shifts of m positions, $m \geq 2$, for m a divisor of N ([6],pp. 256-261). In effect, this decomposes the code into m orbits of n bits, $n = N/m$, and the code is preserved under a cyclic shift $mod\ n$ simultaneously in all m orbits.

The first step in forming a QCRA is to introduce "tailbiting" in the convolutional accumulator code. Instead of view the RA code as the output of a potentially semi-infinite accumulator chain, a chain containing N channel bits and N parity nodes can be closed on itself, forming a loop, with the last channel bit considered to be connected to the first parity node. The start state and the end state of the accumulator are forced to be the same. This is shown for $q = 3,\ N = 15$ QCRA code in Figure 2. For the sake of encoding and decoding, this might seem a nuisance, but it is of little consequence. For q even, there is always a codeword solution for any single non-zero information bit starting and ending in the zero state. The initial state is thus a free variable that can be viewed as an additional information bit. To eliminate the memory requirement, the initial and final state can be set to zero. For q odd, there is no solution for a single non-zero information bit. For the tailbiting RA code to have a solution, the parity of the incoming information bits must be even, guaranteeing that an even number of taps into the accumulator chain are non-zero. By sacrificing one bit, no matter how large K, the initial and final state again becomes a free variable that can be set to zero. With this proviso, the minimum distance of the tailbiting RA code is at least as large as that of the initial RA code inasmuch as any word in the tailbiting RA code must be a word in the original RA code.

A tailbiting RA code is a quasi-cyclic code if the permutation of the inputs exhibits a shift invariance. Specifically, let m be a divisor of N, $mn = N$ and index the parity nodes in the accumulator loop by the integers *mod N*. The code will be invariant under a shift of m positions if the code graph connections are themselves m-shift invariant. That is, if for any information bit v_B that connects to a set of parities with indexes $i_0, i_1, \ldots, i_{q-1}$, there is another information bit v'_B such that v'_B is connected to parities $i_0 + m, i_1 + m, \ldots, i_{q-1} + m$, all indexes *mod N*. The accumulator loop is invariant under any cyclic shift *mod N* (parity to parity, channel bit to channel bit), and therefore for any shift by m, so the code itself is m-shift invariant. With this invariance, the code graph can be redrawn to have a cyclic symmetry, with all information bits and all channel bits appearing in cyclic orbits of size n. For the channel bits, take every mth bit of the accumulator loop for the channel bits to form a cycle of n; for the information bits, take those are mapped to each other by the m-shift invariance. Consequently, the parity check matrix for a QCRA code can be represented as a block matrix of $n \times n$ circulants. Effectively a shift of one position *mod n* in the circulants is equivalent to a shift of m positions *mod N* in the original code.

The parity check matrix H_{sys} for the systematic QCRA code is shown in Figure 2. Let I_s be the $n \times n$ identity matrix cyclically row left-shifted by s positions, a circulant matrix with a single one in the sth row of the first column, $0 \leq s \leq n-1$. There are $N = mn$ rows consisting of m $n \times n$ circulant blocks, and $mn + mn/q$ columns, an $m \times m(q+1)/q$ block matrix of circulants. The channel bits of the accumulator loop are on the right. That each channel bit has degree 2 is reflected by the presence of only two I_s matrices (with either $s = 0$ or $s = 1$) in each block column, giving 2 ones per column. Each row of H_{sys} represents one of the parity checks in the accumulator loop. Each parity must have degree 3, and there are two ones per row due to channel bit connections, leaving one connection to information bits. Thus each row of the block matrix has exactly one I_s matrix, for some s, in the information bit columns. Finally, each information bit has degree q, and therefore each information column of the block matrix contains q shifted identity matrices. Although here we have shown the accumulator loop equations using only I_0 and a single I_1, this is only way of representing the loop. For example, all m matrices on the diagonal can be taken to be I_0 and all the off-diagonal matrices to be I_j. So long as mj is relatively prime to n, the graph will have a single accumulator loop.

4 Recursive Quasi-Cyclic Graph Construction

The challenge of QCRA construction is to choose the shifts of various matrices I_s in H so as to define a code of relatively high quality for which an iterative decoding algorithm will perform well [4]. For the sake of latter, two heuristics are valuable: 1) making the girth g, or shortest cycle, of the graph as large as possible; and 2) making the diameter d, the maximum of the minimum path length between any pair of nodes, as small as possible. A large girth assures that the initial iterations of a sum-product decoding algorithm generate statistically independent estimates. A small diameter assures that information from every received bit will contribute to the probabilistic estimate for any other bit rapidly, after a number of iterations proportional to the diameter. The QCRA construction technique we propose is recursive, defining the code graph for a long code in terms of the code graph of a short code with the intended rate while attempting to keep the diameter as small as possible. The general recursive construction technique, here called a quasi-cyclic or, more precisely, an n-cyclic extension, is applicable to any graph, but our interest is in bipartite code constraint graphs for QCRA codes.

Consider a bipartite graph G with a set of vertices $V = V_p \cup V_b$ consisting of bit vertices and parity vertices with a set of directed edges $E = \{e | e = (v_i, v_j), v_i \in V_p, v_j \in V_b\}$ and incidence matrix H with rows indexed by vertices from V_p and columns indexed by vertices from V_b. Let g and d be the girth and diameter of G, respectively. An *n-cyclic extension* of G, G', is defined as follows. The vertices $V' = \{[v, k] | v \in V, 0 \leq k \leq n-1\}$ are ordered 2-tuples, a vertex of G followed by an integer *mod n*. The edges $E' = \{([v_i, k], [v_j, (k - f(e) mod\ n)]) | (v_i, v_j) = e \in E, 0 \leq k \leq n-1\}$ for some function $f(\cdot)$ defined on the set of edges mapping each edge of G to the integers *mod n*. The function $f(\cdot)$ will be called the *offset function* . In brief, each vertex of G is replaced by n copies, each edge of G by a bundle of n edges, and the offset function $f(\cdot)$ determines a cyclic shift or offset of the bundle that occurs as it passes between the two expanded vertices of G. In matrix terms, the incidence matrix H' of G' is obtained by replacing each "1" in H corresponding to edge e by a matrix I_s, where $s = f(e)$. Graph G' is specified entirely by G, n, and the offset function $f(\cdot)$.

It follows immediately from the specification that each of the n copies of a vertex v has the same degree as v. For example, if G is the graph for an RA code and each parity has degree three, then the parities in G' will also have degree three. It is also easy to prove that the girth satisfies $g' \geq g$, since any closed cycle in G' passes through a series of vertices that project (by dropping the integer component) to a series of vertices forming a closed cycle in G. A cycle in G' is a cycle in G such that a running total computed by subtracting the integer $f(e)$ along an edge from a parity vertex to a bit vertex and adding $f(e)$ along an edge from a bit to a parity is zero *mod n* around the cycle. The goal in extending a graph so as to increase girth is to choose $f(\cdot)$ so that none of the cycles of length g can have a running total of zero *mod n*.

The diameter d' of G' can be much larger than d. Clearly $d' \geq d$, by again projecting: the shortest path in G' between any two vertices $v_1' = [v_1, k_1]$ and $v_2' = [v_2, k_2]$, where v_1 and v_2 are at distance d in G, must be at least d, since its projection is a path in G. The goal in constructing G' to limit the increase in the diameter is to try to ensure that, for an arbitrary pair of vertices of G and any desired offset $k \pmod n$, there is some path of length d' or less between them along which the cumulative offset achieves k. This motivates the investigation of QCRA constructions endowing the graph with a large automorphism group. The automorphism group facilitates the investigation of how all possible cumulative offsets can be achieved along paths between two vertices of G. The automorphism group of an n-cyclic extension has a cyclic subgroup of order n, by design, but it is possible to introduce even stronger fruitful symmetry.

5 QCRA Definition and Subgroups on $GF(p)$

For any prime p, let a be a primitive element of the Galois field $GF(p)$, so that the multiplicative order of a is $p-1$, $a^{p-1} = 1$, and any non-zero elements of the field can be written a^i, $0 \leq i \leq (p-1)$. If q divides $p-1$, $qz = (p-1)$, and a^z is an element of order q, $(a^z)^q = a^{zq} = 1$. The non-zero elements of the field can organized into an array

A with q rows and z columns such that $A(i,j) = a^{iz+j}, 0 \leq i \leq (q-1), 0 \leq j \leq (z-1)$. The entries of A will be used to define the function $f(\cdot)$ in a p-cyclic extension. A is a coset decomposition using a subgroup of order q of the multiplicative group of $GF(p)$.

Let G be a graph for a QCRA code consisting of an accumulator loop containing $p-1$ parity nodes alternating with $p-1$ channel bits. Let the parity vertexes be indexed by the integers *mod* $(p-1)$, for notational clarity written as a function $v_P(i)$, and likewise for channel bits $v_C(i)$ The jth input information bit vertex, $v_B(j)$ of degree q, is connected to parities $v_P(k), k = 0+j, z+j, 2z+j, \ldots, (q-1)z+j$, all indexes *mod* $(p-1)$. This renders the graph shift-invariant under any cyclic shift of the accumulator loop taking parities to parities and channel bits to channel bits. As a starting point, this code appears curiously weak. The girth of the graph is eight, and the minimum distance of the code is q, achieved by making two successive information bits non-zero, no matter how large p. The simple form of G comes into play, however, in showing that a quasi-cyclic extension can be a graph with good connectivity. A p-cyclic extension for $p = 13, q = 3$ and $z = 3$ is illustrated in Figure 3.

Our p-cyclic extension graph of G, G', is the constraint graph for a length $N = nm$ QCRA code, with $n = p$ and $m = p-1$. Critical to the definition, the function $f(e)$ is

$$f((v_P(iz+j), v_B(j))) = A(i,j) = a^{iz+j}, 0 \leq i \leq (q-1), 0 \leq j \leq (z-1) \qquad (1)$$

which can also be written

$$f((v_P(k), v_B(k \, mod \, z))) = a^k, 0 \leq k \leq (p-1) \qquad (2)$$

The offset $f(e)$ for the edge from any parity to any channel bit is zero, except for the 0th parity to $p-2$th channel bit, where $f((v_P(0), v_C(p-2))) = 1$. (We will henceforth assume all index arguments of $v_B(\cdot)$ are *mod* z, all index arguments of $v_P(\cdot)$ and $v_C(\cdot)$ are *mod* $(p-1)$.)

This one anomalous value is needed to form a single accumulator loop with $p(p-1)$ channel bits. Alternatively, any function values that have a net sum around the accumulator loop not divisible by p will suffice.

6 Graph Properties

The diameter of a QCRA graph thus defined depend on the differences in indexes that are induced by the array A. We start by considering the modified graph G'' in which the anomalous value is also set to zero, $f((v_P(0), v_C(p-2))) = 0$, and later compare the diameter of G' to that of G''. G'' has an isolated accumulator loop of length $p-1$ for each value $0 \leq k \leq (p-1)$ and those loops are interconnected solely by the information bits. As for G' the vertices of G'' are $V'' = \{[v,k] | v \in V, 0 \leq k \leq p-1\}$ are ordered 2-tuples, a vertex of G followed by an integer *mod* p. The edges of G'' from parities to information bits are the same; those from parities to channel bits are the same, excepting the anomalous edges $([v_P(0), k], [v_C(p-2), k]), 0 \leq k \leq p-1$ in G''.

Besides the shift invariance in the p-cycles, the automorphism group of G'' also includes a permutation of vertices that is one-transitive on the information nodes.

Proposition: The permutation π that maps $[v_B(j), k]$ to $[v_B(j+1), ak]$, $[v_P(iz+j), k]$ to $[v_P(iz+j+1), ak]$, and $[v_C(iz+j), k]$ to $[v_C(iz+j+1), ak]$, $0 \leq k \leq p-1$, preserves the edges of G''.

Proof: The offset function $f(e)$ has been chosen so that cyclic shift in G accompanied by multiplication of all offset values by a leaves the graph and offset values invariant.

There is only one non-trivial condition to check. Repeating π z times maps $[v_B(j), k]$ to $[v_B(j), a^z k]$, and maps $[v_P(iz+j), k]$ to $[v_P((i+1)z+j), a^z k]$. By the definition above, $(v_P((i+1)z+j, v_B(j)))$ is an edge in G and $f((v_P((i+1)z+j), v_B(j)) = a^{(i+1)z+j} = a^z a^{iz+j}$ as required for consistency. The parity to channel bit edges are also preserved by π, and the offset function for those edges is zero per the definition of G'', so multiplying by a preserves the function values. Therefore, π is an automorphism of G''.

The p-cyclic extension ensures that, by definition, the shift σ mapping each node $[v(j), k]$ to $[v(j), k+1]$ is an automorphism of G''. Together σ and π are one-transitive on information bit vertexes and one-transitive on parity vertexes and on channel bit vertexes. To bound the diameter of the graph, we will start by bounding the length of the shortest path between two parity nodes. Our definition of the array A permits a quick but weak upper bound to be established, and we now sketch the argument. In G'' any node $[v_P(j), k]$ can be connected to any other parity node with the same k by a path of length less than $(p-1)$, because both nodes are part of an accumulator loop of length $2(p-1)$. We will exhibit one path from a node $[v_P(j), k]$ to another arbitrary $[v_P(j'), k+\Delta]$. Observe that $[v_P(j), k]$ is connected via $[v_B(j \ mod \ z), k]$ to $[v_P(iz+j), k + a^j(a^{iz}-1)]$ for $1 \leq i \leq (q-1), 0 \leq j \leq (p-2), 0 \leq k \leq (p-1)$. For each of the $(q-1)$ non-zero values of i, the equation $a^j(a^{iz}-1) = \Delta$ has a unique solution for j, $j = log_a(\Delta) - log_a(a^{iz}-1)$, the discrete logarithm on $GF(p)$, for any non-zero difference Δ. Thus there are $(q-1)$ distinct paths of length two connecting nodes in the set $\{[v_P(j), k] | 0 \leq j \leq (p-2)\}$ to nodes in the set $\{[v_P(j), k+\Delta] | 0 \leq j \leq (p-2)\}$ for any $0 \leq \Delta \leq p-1$. Moreover, stated non-rigorously, the discrete logarithm gives a pseudo-random distribution to j for $1 \leq i \leq (q-1)$, which helps avoid the formation of a large number of short cycles in the graph.

The offset $f(e)$ was chosen purposefully to give the quasi-cyclic extension this dispersion property. Any $(p-1)$ parity node segment of the accumulator loop of G' is multiply connected to any other $(p-1)$ parity node segment. In this sense, the offset function induces a graph similar to that of a difference set in cyclic coding theory [5](pp. 397-98); a rich set of differences is created between information bit tap connections, causing information to propagate rapidly in an iterative decoding process. With this alone we can give a weak upper bound on graph diameter.

Proposition: The diameter of the quasi-cyclic extension graph G' satisfies
$d \leq 4(p-1)+4$.

Proof: First, in G'' any parity node $[v_P(j), k]$ is at most $2(p-1)+2$ steps from any other $[v_P(j'), k']$, as follows: There is a path along the accumulator loop of length $(p-1)$ or less (counting both channel bit nodes and parity nodes) going from $[v_P(j), k]$ to a $[v_P(j_1), k]$ which is connected via a single information bit to a parity $[v_P(j_2), k']$, by the preceding discussion. (In fact, there are $(q-1)$ choices for $v_P(j_1)$.) Then there is a path, again of length $(p-1)$ or less, from $[v_P(j_2), k']$ to $[v_P(j'), k']$. The concatenated path has length $2(p-1)+2$ or less. In the actual quasi-cyclic extension graph G', the accumulator loop is not broken up into length $2(p-1)$ segments; rather, it is one unbroken chain. However, there exist a path in G' from $[v_P(j), k]$ to a $[v_P(j_1), k]$ of length at most $2(p-1)$ that does not cross one of the segment boundaries, with $[v_P(j_1), k]$ again connected via a single information bit to a parity $[v_P(j_2), k']$, and then a path of length at most $2(p-1)$ that does not cross a segment boundary to $[v_P(j'), k']$, a total length of at most $4(p-1)+2$. This path is the same in both G' and G''. Finally, any channel node is distance one from a parity node, as is any information bit node, which means the diameter can be at most two greater.

This bound is very weak because it does not exploit the shortest of the $q-1$ paths between segments that are available, and it ignores all paths that pass through multiple information bit nodes. For large q, these paths become very dense rapidly and should dominate in determining both the diameter and the girth of the quasi-cyclic extension graph. Here again the automorphism group of G'' is a powerful tool for analyzing the density of multiple information bit paths, entailing a study of the differences in the p-cyclic second component of a parity node that can be achieved in two or more steps.

As mentioned previously, the girth of G' is the shortest cycle in G along which the cumulative sum of the offset function values is zero. Many paths with cumulative sum zero can be found that repeat any fixed cycle in G p times, thus arriving at zero $mod\ p$. There are also sum zero paths obtained by passing along the same edges of G twice, in opposite directions, but these generally involve traverses of $O(p)$ along the accumulator loop of G. The intricate analysis studies the differences between the values in any column of the array A and the differences of the differences between immediately adjacent columns, and so forth for additional steps, until two distinct paths from the same starting bit of G are found to arrive at the same vertex of G with the same cumulative sum. The group property of the columns of A seems to avoid immediate short cycles, but we have no general proof that addresses the potentially subtle number theoretic questions involved.

7 Minimum Distance Arguments

One objective in describing a QCRA code algebraically is to achieve a weight distribution for the code that is superior to those resulting from those found in RA codes with random permutations. In particular, a lower bound on the minimum distance of the code would give assurance that an error floor will not be encountered prematurely at high signal to noise ratios.

Our p-cyclic QCRA codes are amenable to transform analysis over an extension field $GF(2^x)$ containing a pth root of unity, leading to a characterization of the code space as a set of $(p-1)$ interrelated cyclic codes of length p [7]. This representation can be the source of an algebraic bound on the minimum distance of the code. For example, it can be shown algebraically that the rate 1/3 length 15 code of Figure 2 has a minimum distance of 7 and is equivalent to a [15,5,7] BCH code. To date we have not succeeded in deriving a general bound that can be readily computed for arbitrary p, and with the page limits on these Proceedings in mind, we will only make a few brief observations that give insight into the design principles.

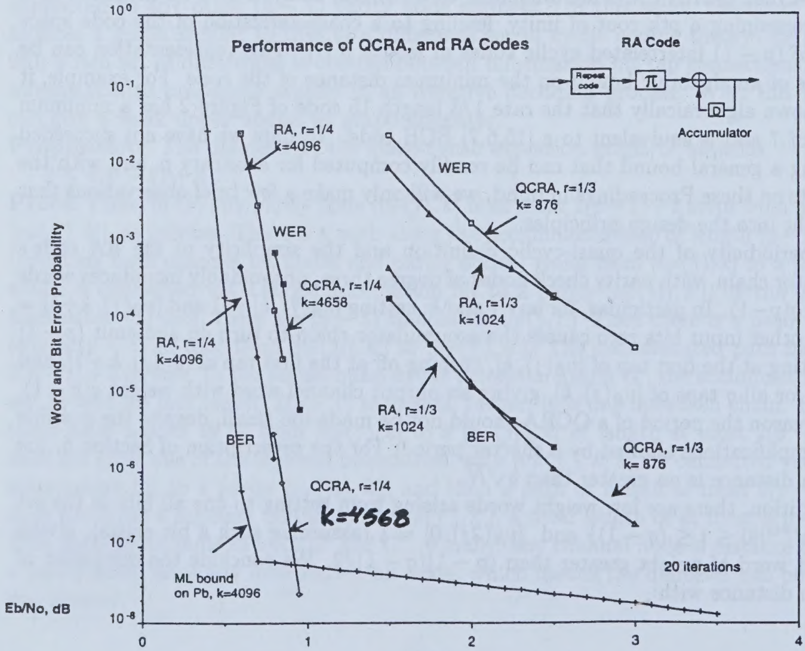
The periodicity of the quasi-cyclic definition and the simplicity of the RA code's accumulator chain, with parity check nodes of degree three, unavoidably introduces words of weight $q(p-1)$. In particular, for any j and k, setting $[v_B(j), k] = 1$ and $[v_B(j), k+1] = 1$ and all other input bits zero causes the accumulator chain to turn on and emit $(p-1)$ ones starting at the first tap of $[v_B(j), k]$, turning off at the first tap of $[v_B(j), k+1]$, and similarly for all q taps of $[v_B(j), k]$, giving an output channel word with weight $q(p-1)$. For this reason the period of a QCRA should not be made too small, despite the possible circuit simplifications allowed by a shorter period. For the prescription of Section 5, the minimum distance is no greater than $q\sqrt{N}$.

In addition, there are low weight words arising from setting to one all bits in the set $\{[v_B(j), a^{j+iz}] | 0 \leq i \leq (q-1)\}$ and $[v_B(2j), 0] = 1$ (assuming such a bit exists), giving an output word of weight greater than $(p-1)(q-1)/2$. We conclude the discussion of minimum distance with:

Conjecture: The minimum distance of the $[(p-1)p, (p-1)p/q, D_{min}]$ QCRA code of Section 5 satisfies $(p-1)(q-1)/2 \leq D_{min} \leq (p-1)q$.

8 Simulation Results

Two QCRA codes were simulated on the Additive White Gaussian Noise (AWGN) channel by Dariush Divsalar of NASA JPL, Pasadena, and compared with RA codes of the same rate and roughly similar length. The permutation used in the comparison codes were generated at random, then screened to eliminate permutations for which the information bit taps did not pass a "spread" or "dispersion" test. The shorter QCRA is a rate 1/3 of length $N = 2638 = 36 \times 73$ with $K = 876$ information bits, a variant of the prescription of Section 5 in which only columns 0 through 35 of the A matrix were used to define information bits in a 73-cyclic extension of a length 36 code. The second was a rate 1/4 with $p = 137$ and $N = 18632 = 137 \times 136, K = 4658$. All codes were decoded with an iterative decoder of the turbo style. The results are shown in Figure 4. Compared to a somewhat longer $K = 1024$ RA code, and with 20 decoder iterations, the QCRA has higher Word Error Rate (WER) and Bit Error Rate (BER) at lower E_b/N_0 but outperforms the "random permutation" RA by both measures at higher E_b/N_0. We conjecture that this behavior can be attributed to a better minimum distance for the shorter QCRA. The longer $K = 4658$ QCRA code is about 0.1 db worse in WER than its somewhat shorter RA competitor, and virtually the same in BER. Of particular note is a point at 2.05×10^{-8} that lies beneath a lower bound on the BER of the ensemble of random codes with maximum likelihood decoding. The data point represents the decoding of 1.705×10^6 blocks, a total of 3.177×10^{10} bits, with 163 uncorrected information bit errors out of 7.942×10^9 information bits, at $E_b/N_0 = 0.95$. Eight additional iterations reduced the number of bit errors further, from 163 to 2.



9 Conclusion

The use of random permutations in turbo style codes is a commonly accepted technique. Despite their numerous drawbacks, random permutations, or rather, carefully screened and selected random permutations, create a code constraint graph that is known to have a low density of short cycles and good performance with iterative decoders. This paper presents one algebraic description for a special class of very simple turbocodes, the repeat-accumulate codes, using an apparently novel recursive graph construction technique. These algebraic RA codes are quasi-cyclic, being invariant with respect to simultaneous shifts of multiple cycles of length p. Effectively, the algebra defines the permuted connections of the information bits in an RA code. The definition was devised so as to make the code constraint graph well connected, within the limitation of the very restricted degree of the RA code graph nodes. While we have not yet established a meaningful bound on the minimum distance of the codes, simulation studies indicate that our QCRA codes can have a significantly larger minimum distance than their random permutation competitors. RA codes themselves do not achieve the energy efficiencies possible with much more complex turbocode systems, but their simplicity may outweigh the energy advantage in some applications. The import of our arguments is that algebraic methodologies can be fruitful for the design of codes for iterative decoding systems that are superior to those that use purely random graph constructions.

References

[1] S. Benedetto and G. Montorsi, "Unveiling Turbo Codes: Some Results on Parallel Concatenated Coding Schemes," *IEEE Trans. on Information Theory*, vol. 42, no. 2, pp. 409-428, March 1996.

[2] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon limit error-correcting coding and decoding: turbo codes,," *Proc. 1993 IEEE International Conference on Communications*, Geneva, Switzerland, pp. 1064-1070, May 1993.

[3] Dariush Divsalar, Hui Jin, and Robert J. McEliece, "Coding Theorems for "Turbo-like Codes," *Proceedings 36th Allerton Conf. Commun., Control, and Computing*, pp. 201-210, September 1998.

[4] F. Kschischang and B. Frey, "Iterative Decoding of Compound Codes by Probability Propagation in Graphical Models," *IEEE Jour. Selected Areas in Communications*, vol. 16, no. 2, pp. 219-230, February 1998.

[5] F. J. MacWilliams and N. J. A. Sloane, *The Theory of Error-Correcting Codes*, North-Holland,Second Printing, 1978.

[6] W. W. Peterson and E. J. Weldon,Jr., *Error-Correcting Codes*,Second Edition, MIT Press, 1972.

[7] R. M. Tanner, "A Transform Theory for a Class of Group-Invariant Codes," *IEEE Trans. on Information Theory*, vol. 34, no. 4, pp. 752-775, July 1988.

The Serial Concatenation of Rate-1 Codes Through Uniform Random Interleavers*

Henry D. Pfister and Paul H. Siegel
Signal Transmission and Recording Group
Department of Electrical and Computer Engineering
University of California, San Diego
La Jolla, California 92093-0407
{hpfister, psiegel} @ucsd.edu

Abstract

Until the Repeat Accumulate codes of Divsalar, *et al.* [4], few people would have guessed that simple rate-1 codes could play a crucial role in the construction of "good" codes. In this paper, we will construct "good" linear block codes at any rate $r < 1$ by serially concatenating an arbitrary outer code of rate r with a large number of rate-1 inner codes through uniform random interleavers. We derive the average output weight enumerator for this ensemble in the limit as the number of inner codes goes to infinity. Using a probabilistic upper bound on the minimum distance, we prove that long codes from this ensemble will achieve the Gilbert-Varshamov bound with high probability. Finally, by numerically evaluating the probabilistic upper bound, we observe that it is typically achieved with a small number of inner codes.

1 Introduction

The introduction of turbo codes by Berrou, Glavieux, and Thitimajshima [3] is remarkable because it combined simple components together to set a new standard for error-correcting codes. Since then, iterative "turbo" decoding has made it practical to consider a whole new world of concatenated codes while the use of "random" interleavers and recursive convolutional encoders has given us a starting point for choosing new code structures. Many of these concatenated code structures fit into a class that Divsalar, Jin, and McEliece call "turbo-like" codes [4]. This class includes their Repeat Accumulate (RA) codes which consist only of a repetition code, an interleaver, and an accumulator. Still they prove that, for sufficiently low rates and any fixed E_b/N_0 greater than a threshold, these codes have vanishing word error probability as the block length goes to infinity. This shows that powerful error-correcting codes may be constructed from extremely simple components.

In this paper we consider the serial concatenation of an arbitrary outer code of rate $r < 1$ with m identical rate-1 inner codes where, following the convention of turbo coding literature, we use the term serial concatenation to mean serial concatenation through a "random" interleaver. Any real system must, of course, choose a particular interleaver. Our analysis, however, will make use of the *uniform random interleaver* (URI) [2] which is equivalent to averaging over all possible interleavers. We analyze this system using a probabilistic bound on the minimum distance and show that, in the limit as the number of inner codes m goes to infinity, the minimum distance is bounded by an expression that resembles the Gilbert Bound (GB) [5].

---

*This work was supported in part by the National Science Foundation (NSF) under grant NCR-9612802 and by the National Storage Industry Consortium (NSIC).

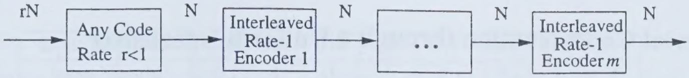


Figure 1: Our system consists of any rate $r < 1$ code followed by m rate-1 codes.

Our work is largely motivated by [4] and by the results of Öberg and Siegel [10]. Both papers consider the effect of a simple rate-1 "Accumulate" code in a serially concatenated system. In [4] a coding theorem is proved for RA codes, while in [10] the "Accumulate" code is analyzed as a precoder for the dicode magnetic recording channel. Benedetto, *et al.* also investigated the design and performance of Double Serially Concatenated Codes in [1].

If the outer code consists of multiple independent copies of a short block code and the inner code is a cascade of m interleaved "Accumulate" codes, we will refer to these codes as Generalized Repeated Accumulated (GRA^{m}) codes. McEliece has analyzed the maximum likelihood decoding performance of these codes for $m = 1$ [9], and we focus on the minimum distance of these codes for $m \geq 1$.

The outline of the paper is as follows. In Section 2 we review the *weight enumerator* (WE) of linear block codes and the union bound on the probability of error for maximum likelihood decoding. We also review the average weight enumerator for the serial concatenation of two linear block codes through a URI, and relate serial concatenation to matrix multiplication using a normalized form of each code's *input output weight enumerator* (IOWE). In Section 3 we introduce our system, shown in Figure 1, and we compute its average output WE. In Section 4 we derive a probabilistic bound on the minimum distance of any code, taken from a random ensemble, in terms of the ensemble's average WE. Applying this bound to the WE from Section 3 gives an expression very similar to the GB, and examining the bound as the block length goes to infinity produces the Gilbert-Varshamov Bound (GVB). In Section 5 we numerically evaluate our bound on minimum distance for various GRA^{m} codes and observe that 3 or 4 "Accumulate" codes seem to be sufficient to achieve the bound corresponding to asymptotically large m. Finally, in Section 6 we discuss some conclusions and directions for future work.

2 Weight Enumerators and Serial Concatenation

2.1 The Union Bound

In this section, we review the weight enumerator of a linear block code and the union bound on error probability for maximum likelihood decoding. The IOWE $A_{w,h}$ of an (n,k) block code is the number of codewords with input weight w and output weight h, and the WE A_h is the number of codewords with output weight h and any input weight. Using these definitions, the probability of word error is upper bounded by

$$P_w \leq \sum_{h=1}^{n} \sum_{w=1}^{k} A_{w,h} z^h,$$

and the probability of bit error is upper bounded by

$$P_b \leq \sum_{h=1}^{n} \sum_{w=1}^{k} \frac{w}{k} A_{w,h} z^h.$$

The term z^h represents an upper bound on the pairwise error probability, between any two codewords differing in h positions, for the channel of interest. The constant z is defined for many memoryless channels [7, Section 5.3], and for the AWGN channel it is $z = e^{-(k/n)(E_b/N_0)}$.

2.2 Serial Concatenation through a Uniform Interleaver

In this section, we review the serial concatenation of codes through a uniform random interleaver. The introduction of URI in the analysis of turbo codes by Benedetto and Montorsi [2] has made the analysis of complex concatenated coding systems relatively straightforward, and using the URI for analysis is equivalent to averaging over all possible interleavers. The important property of the URI is that the distribution of output sequences is a function only of the weight distribution of input sequences. More precisely, an input sequence of weight w produces all possible output sequences of weight w, each with equal probability.

Consider any (n, k) block code with IOWE $A_{w,h}$ preceded by a URI. We will refer to such a code as a *uniformly interleaved code* (UIC). The probability of the combined system mapping an input sequence of weight w to an output sequence of weight h is

$$Pr\,(w \rightarrow h) = \frac{A_{w,h}}{\binom{k}{w}}. \tag{1}$$

We can now consider an (n, k) block code formed by first encoding with an (n_1, k) outer code with IOWE $A^{(o)}_{w,h}$, then permuting the output bits with a URI, and finally encoding again with an (n, n_1) inner code with IOWE $A^{(i)}_{w,h}$. The average number of codewords with input weight w and output weight h is then given by

$$\begin{aligned}\overline{A}_{w,h} &= \sum_{h_1=0}^{n_1} A^{(o)}_{w,h_1}\, Pr\,(h_1 \rightarrow h) \\ &= \sum_{h_1=0}^{n_1} A^{(o)}_{w,h_1} \frac{A^{(i)}_{h_1,h}}{\binom{n_1}{h_1}}.\end{aligned} \tag{2}$$

The average IOWE for the serial concatenation of two codes may also be written as the matrix product of the IOWE for the outer code and a normalized version of the IOWE for the inner code. Let us define, for any code, the *input output weight transition probability* (IOWTP) $P_{w,h}$ as the probability that a uniform random input sequence of weight w is mapped to an output sequence of weight h. From (1), we can see that

$$P^{(i)}_{w,h} = \frac{A^{(i)}_{w,h}}{\binom{k}{w}}. \tag{3}$$

Substituting (3) into (2), we have

$$\overline{A}_{w,h} = \sum_{h_1=0}^{n_1} A^{(o)}_{w,h_1} P^{(i)}_{h_1,h} = \mathbf{A}^{(o)}\mathbf{P}^{(i)}.$$

where $\mathbf{A}^{(o)}$ is the matrix representation of the outer code IOWE and $\mathbf{P}^{(i)}$ is the matrix representation of the inner code IOWTP. By inductively applying this to multiple inner code IOWTP matrices, one can see that matrix multiplication computes the overall $\overline{A}_{w,h}$ for an arbitrary number of serial concatenations. It is also clear from (3) that IOWTP matrices are stochastic (i.e. all rows sum to 1).

2.3 A Simple Example

In this section, we will compute the IOWE and IOWTP of the rate-1 "Accumulate" code [4]. The "Accumulate" code is a block code formed by truncating the simplest recursive convolutional code possible, having generator matrix $G(D) = 1/(1 \oplus D)$, after n symbols. The

Input Sequence	000	001	010	100	011	101	110	111
Input Weight	0	1	1	1	2	2	2	3
Output Sequence	000	001	011	111	010	110	100	101
Output Weight	0	1	2	3	1	2	1	2

Table 1: Input-output sequences and weight mappings for $n = 3$ "Accumulate" code.

generator matrix for this block code is an $n \times n$ matrix with all 1's in the upper triangle and all 0's elsewhere. In the example, we will look at the case $n = 3$. The generator matrix is

$$G = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix}.$$

Using Table 1, we see that a uniform random input of weight 1 maps to output weights 1, 2, and 3 with equal probability, and cannot be mapped to output weight 0. So the $w = 1$ row of the IOWTP matrix is $\left[\begin{array}{cccc} 0 & 1/3 & 1/3 & 1/3 \end{array}\right]$. Filling in the rest of the entries, we give both the IOWE $A_{w,h}$ and the associated IOWTP $P_{w,h}$ in matrix form:

$$A_{w,h} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 2 & 1 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}_{w,h}, \quad P_{w,h} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1/3 & 1/3 & 1/3 \\ 0 & 2/3 & 1/3 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}_{w,h}.$$

3 Multiple Rate-1 Serial Concatenations

3.1 The Input Output Weight Enumerator

In this section, we will consider a code formed by encoding $m + 1$ times. The first (outer) encoder is for an (n, k) block code with IOWE $A^{(o)}_{w,h}$. The next m (inner) encoders are for identical rate-1 UICs of block length n with IOWE $A^{(i)}_{w,h}$. If we let $\mathbf{P}$ be the IOWTP matrix associated with $A^{(i)}_{w,h}$, then we can write the average IOWE $\overline{A}_{w,h}$ for this code as

$$\overline{A}_{w,h} = \sum_{h_1=0}^{n} A^{(o)}_{w,h_1} \left[\mathbf{P}^m\right]_{h_1 h}. \tag{4}$$

The linearity of the code guarantees that the matrix $\mathbf{P}$ will be block diagonal with at least two blocks because inputs of weight 0 will always be mapped to outputs of weight 0 and inputs of weight greater than 0 will always be mapped to outputs of weight greater than 0. So let the first block be the 1×1 submatrix associated with $w = h = 0$, and let the second block $\mathbf{Q}$ be the $n \times n$ submatrix formed by deleting the first row and column of $\mathbf{P}$. Writing $\mathbf{P}^m$as the product of block diagonal matrices, we see that

$$\mathbf{P}^m = \begin{bmatrix} 1 & 0 \\ 0 & \mathbf{Q}^m \end{bmatrix}.$$

3.2 Stationary Distributions and Markov Chains

In this section, we will discuss the stationary distributions of a Markov Chain (MC) and how they relate to the stationary weight distributions of a rate-1 UIC. This discussion is based on the observation that if $\mathbf{P}$ is a finite dimensional stochastic matrix, then there is an associated MC with state transition matrix $\mathbf{P}$. Applying this to the IOWTP matrix of any UIC, we see that all UICs have an associated MC.

A MC, with state transition matrix $\mathbf{P}$, has a stationary distribution $\pi = [\pi_0, \ldots, \pi_n]$ if $\pi\mathbf{P} = \pi$ and $\sum \pi_i = 1$. Accordingly, a rate-1 UIC has a stationary weight distribution π if π is a stationary distribution of the code's associated MC. Recall that a MC is *irreducible* if there is a path from any state to any other state with a finite number of steps. Using these definitions, we can draw upon some well-known results from the theory of non-negative matrices and MCs [11].

THEOREM 1. *An irreducible Markov Chain has a unique stationary distribution.*

We define a rate-1 UIC to be *irreducible* if the $\mathbf{Q}$ submatrix of its IOWTP matrix $\mathbf{P}$ can be associated with an irreducible MC. Similarly, this implies that there is a path from any weight to any other weight in a finite number of encodings. We now apply Theorem 1 to the $\mathbf{Q}$ submatrix of an irreducible rate-1 UIC. Since the matrix $\mathbf{Q}$ does not include inputs and outputs of weight 0, we must assume $\pi_0 = 0$ to make the following stationary weight distribution unique.

PROPOSITION 1. *The unique stationary weight distribution $\pi = [\pi_0, \ldots, \pi_n]$ of an irreducible rate-1 UIC with $\pi_0 = 0$ is*

$$\pi_h = \frac{\binom{n}{h}}{2^n - 1} \text{ for } 1 \le h \le n.$$

The example from Section 2.3 is irreducible, and applying Proposition 1 gives

$$\begin{bmatrix} 0 & \frac{3}{7} & \frac{3}{7} & \frac{1}{7} \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1/3 & 1/3 & 1/3 \\ 0 & 2/3 & 1/3 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} = \begin{bmatrix} 0 & \frac{1+2}{7} & \frac{1+1+1}{7} & \frac{1}{7} \end{bmatrix}.$$

An irreducible MC is *primitive* if its state transition matrix has a unique eigenvalue of maximum modulus. Accordingly, we define an irreducible rate-1 UIC to be *primitive* if the MC associated with the $\mathbf{Q}$ submatrix of the IOWTP matrix is primitive. The following theorem from the theory of MCs [11] will allow to examine the asymptotic behavior of (4) as m goes to infinity.

THEOREM 2. *A primitive Markov Chain with state transition matrix $\mathbf{P}$ and unique stationary distribution π satisfies the limit*

$$\lim_{m\to\infty} \mathbf{P}^m = \begin{bmatrix} \pi \\ \vdots \\ \pi \end{bmatrix}.$$

The example from Section 2.3 is also primitive, and applying Theorem 2 gives

$$\lim_{m\to\infty} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1/3 & 1/3 & 1/3 \\ 0 & 2/3 & 1/3 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}^m = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 3/7 & 3/7 & 1/7 \\ 0 & 3/7 & 3/7 & 1/7 \\ 0 & 3/7 & 3/7 & 1/7 \end{bmatrix}.$$

3.3 Asymptotic Behavior for Many Concatenations

In this section, we use (4) and Theorem 2 to compute the average WE of any rate $r < 1$ outer code serially concatenated with m primitive rate-1 UICs, in the limit as m goes to infinity. The intriguing part of this result is that the WE is independent of the particular outer code and primitive rate-1 UIC chosen. We note that this is essentially a new construction for a uniform random ensemble of linear codes.

THEOREM 3. *Consider a rate-1 code formed by serially concatenating m primitive rate-1 UICs. For non-zero input weights and in the limit as m goes to infinity, the output weight distribution is independent of the input weight distribution and is*

$$\pi_h = \frac{\binom{n}{h}}{2^n - 1}.$$

COROLLARY. *The ensemble averaged WE for non-zero output weights for any rate $r < 1$ code serially concatenated with m primitive UICs, in the limit as m goes to infinity, is*

$$\begin{aligned} \overline{A}_h &= \sum_{w=1}^{rn} \sum_{h_1=1}^{n} A^{(o)}_{w,h_1} \left[\mathrm{P}^m\right]_{h_1 h} \\ &= \left(\sum_{w=1}^{rn} \sum_{h_1=1}^{n} A^{(o)}_{w,h_1} \right) \frac{\binom{n}{h}}{2^n - 1} = \left(2^{rn} - 1\right) \frac{\binom{n}{h}}{2^n - 1}. \end{aligned} \tag{5}$$

4 Bounds on the Minimum Distance

4.1 A General Bound on the Distribution of $\mathrm{d}_{\min}$ from $\overline{\mathrm{A}}_\mathrm{h}$

In this section, we derive an upper bound on the probability that a randomly chosen code from some ensemble has $d_{min} < d$. This upper bound can be computed using only the average WE of the ensemble. A similar bound was used by Gallager to bound the minimum distance of Low Density Parity Check Codes [6].

THEOREM 4. *The probability that a code, randomly chosen from an ensemble with average WE $\overline{A}_h$, has $d_{min} < d$ is bounded by*

$$Pr(d_{min} < d) \leq \sum_{h=1}^{d-1} \overline{A}_h.$$

Proof. Let an ensemble of linear codes with average WE $\overline{A}_h$ be defined by a set of WEs $\left\{ A_h^{(1)}, A_h^{(2)}, \ldots, A_h^{(M)} \right\}$ each chosen with equal probability. Further, let $d_{min}^{(i)}$ be the minimum distance of the code associated with $A_h^{(i)}$. We can upper bound the probability of choosing a code with $d_{min} < d$ from this ensemble. First we define an indicator function

$$I(\text{condition}) = \begin{cases} 0 & \text{if condition false} \\ 1 & \text{if condition true} \end{cases}$$

and we note that for all non-negative integers x,

$$I(x > 0) \leq x. \tag{6}$$

First counting the number of codes with $d_{min} < d$, and then substituting an equivalent condition in the indicator function we have

$$Pr(d_{min} < d) = \frac{1}{M}\sum_{i=1}^{M} I(d_{min}^{(i)} < d) = \frac{1}{M}\sum_{i=1}^{M} I\left(\left(\sum_{h=1}^{d-1} A_h^{(i)}\right) > 0\right).$$

Upper bounding the indicator function with (6) and then summing over i gives

$$Pr(d_{min} < d) \leq \frac{1}{M}\sum_{i=1}^{M}\sum_{h=1}^{d-1} A_h^{(i)} = \sum_{h=1}^{d-1} \overline{A}_h.$$

□

4.2 An Application of the Bound

We now apply Theorem 4 to the WE in (5). This leads to a statement that, for a given block length n and rate r, upper bounds the probability of picking a code with minimum distance less than some threshold. Let $d^*(n, r, \epsilon)$ be the largest d which satisfies

$$\sum_{h=0}^{d-1}\binom{n}{h} \leq \frac{2^n - 1}{2^{rn} - 1}\epsilon + 1 \tag{7}$$

for block length n, rate r, and $0 \leq \epsilon \leq 1$. For this d^*, we can rearrange terms to get

$$\sum_{h=0}^{d^*-1}\binom{n}{h} - 1 \leq \frac{2^n - 1}{2^{rn} - 1}\epsilon.$$

Changing the lower limit of the sum and rearranging we have

$$\sum_{h=1}^{d^*-1}\frac{2^{rn} - 1}{2^n - 1}\binom{n}{h} \leq \epsilon.$$

Substituting for the expression of the WE $\overline{A}_h$ given in (5), we have

$$\sum_{h=1}^{d^*-1} \overline{A}_h \leq \epsilon$$

which, when combined with Theorem 4, implies that

$$P(d_{min} < d^*) \leq \epsilon. \tag{8}$$

So with probability at least $1 - \epsilon$, a randomly chosen code from this ensemble will have minimum distance at least $d^*(n, r, \epsilon)$.

The Gilbert Bound (GB) for binary codes [5] says that there exists at least one code with block length n, rate r, and minimum distance d if

$$2^{rn}\sum_{h=0}^{d-1}\binom{n}{h} \leq 2^n. \tag{9}$$

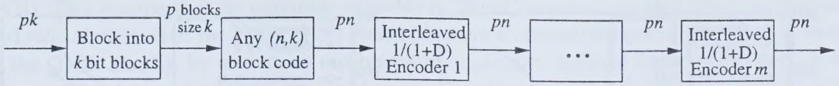


Figure 2: Encoder for a GRA$^{\mathrm{m}}$ Code with the block size indicated at each stage.

If we substitute $\epsilon = (2^{(1-r)n} - 1)(2^{rn} - 1)/(2^n - 1)$ into (7), then we have an expression identical to (9). Since this ϵ is strictly less than one, it follows from (8) that there exists at least one code in our ensemble with $d_{min} \geq d$. So we have qualitatively the same result as the GB.

The Gilbert-Varshamov Bound (GVB) takes its name from the GB and from a related bound due to Varshamov [5]. The GVB is the form of both bounds in the limit as n goes to infinity, and it says that there is a code with rate r and normalized minimum distance $\delta = d_{min}/n$ if

$$H(\delta) \leq 1 - r \tag{10}$$

where $H(x) = -x \log_2 x - (1-x)\log_2(1-x)$ is the binary entropy function.

If we let

$$\delta^*(r, \epsilon) = \lim_{n \to \infty} \frac{1}{n} d^*(n, r, \epsilon)$$

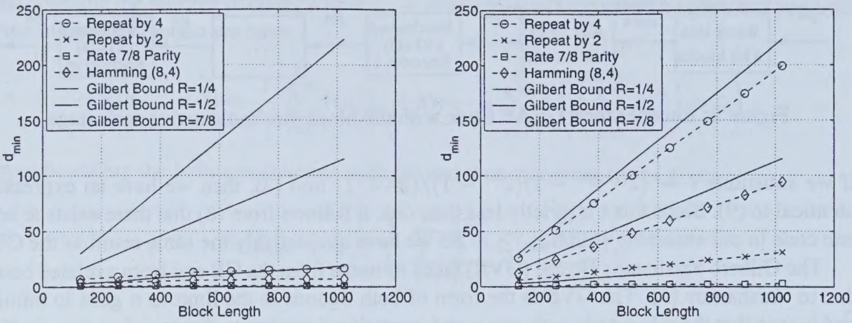
and examine (7) in the limit as n goes to infinity, we find that our bound says something even stronger than the GVB. In fact, we find that $\delta^*(r, \epsilon)$ is equal to the largest δ that satisfies (10) for any $\epsilon > 0$. This implies that in the limit as n goes to infinity, almost all of our codes will have a normalized minimum distance δ satisfying (10). This makes our codes "good" in the sense that, for a fixed rate as the block length goes to infinity, almost all of the codes in our ensemble have a normalized minimum distance that is bounded away from zero. It should be noted that this behavior is well-known for long random codes.

5 Generalized Repeated Accumulated (GRA$^{\mathrm{m}}$) Codes

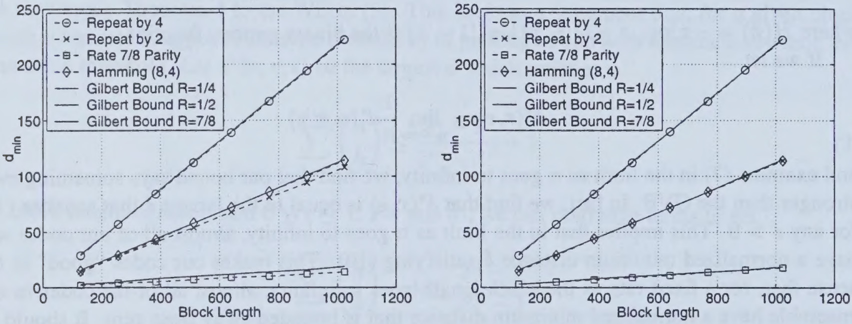
In this section, we describe GRA$^{\mathrm{m}}$ codes and apply Theorem 4 to some specific examples. GRA$^{\mathrm{m}}$ codes are formed by the serial concatenation of a simple outer code, which consists of p independent copies of a short (n, k) block code, and m interleaved rate-1 "Accumulate" codes. The encoder for GRA$^{\mathrm{m}}$ codes is shown in Figure 2. The performance of long GRA^1 codes with maximum likelihood decoding was reported in [9], but cases with $m > 1$ were not considered. So we give results pertaining to the minimum distance GRA$^{\mathrm{m}}$ codes using a few examples.

In order to apply Theorem 4 to a specific ensemble, we must compute its average WE and choose an ϵ. For the following results, we computed the average WEs numerically and chose $\epsilon = 1/2$. This means that at least half of the codes in our ensemble have a minimum distance at least as large as the values shown in Figure 3. For the short block codes, we chose: a repeat by 2 (R2), a repeat by 4 (R4), a rate 7/8 single parity check (P8), and the $(8, 4)$ extended Hamming code (H8).

It is important to note that, at a fixed rate, a "good" code is defined by a minimum distance which grows linearly with the block length. When examining these results, we will focus on whether or not the minimum distance appears to be growing linearly with block length and how close it is to the GB. For $m = 1$, it is known that the typical minimum distance grows $O(n^{(d^o-2)/d^o})$ where d^o is the minimum distance of the repeated outer code [8]. Examining Figure 3 for $m = 1$, we see that the minimum distance grows slowly for R4 and H8 and not all for R2 and P8. While for $m = 2$, the minimum distance growth of R4, H8, and R2



(a) $m = 1$ (b) $m = 2$



(c) $m = 3$ (d) $m = 4$

Figure 3: Probabilistic lower bound on the minimum distance of various GRA$^{\mathrm{m}}$ codes.

appears distinctly linear. It is difficult to determine the growth rate of P8 with $m = 2$ from these results. At $m = 3$, all of the codes appear to have a minimum distance growing linearly with block length and the rates are very close to the GB. Finally, with $m = 4$, the bound on minimum distance and the GB are almost indistinguishable. These results are very encouraging and suggest that, over a wide range of rates, only a few "Accumulate" codes are sufficient to approach the GB on minimum distance.

6 Conclusions and Future Work

In this paper, we began by showing the relationship between serial concatenation through a uniform random interleaver and matrix multiplication of input output weight transition probability (IOWTP) matrices. We then introduced an ensemble of codes consisting of any rate $r < 1$ outer code followed by an infinite number of rate-1 primitive uniformly interleaved codes, and computed the ensemble's average weight enumerator. This was done by introducing a correspondence between IOWTP matrices and Markov Chains (MCs), and drawing on some well-known limit theorems from MC theory. Next, we derived a probabilistic bound on the minimum distance of codes from this ensemble and noted that this bound is almost identical to both the finite block length Gilbert Bound (GB) and the infinite block length Gilbert-Varshamov Bound

(GVB). This implies that the ensemble of codes is "good" because, for long block lengths and fixed rate, almost all of the codes in our ensemble have a normalized minimum distance meeting the GVB. Finally, by evaluating our bound on minimum distance for specific outer codes and a small number of "Accumulate" codes, we observed that a small number of inner codes may be sufficient to approach the bound for an infinite number.

We are currently evaluating the iterative decoding of GRA^m codes and working to prove a coding theorem similar to [4] for these codes.

Acknowledgement. The authors would like to express their gratitude to M. Öberg for posing the initial question that led to this research and for helpful discussions along the way.

References

[1] S. Benedetto, D. Divsalar, G. Montorsi, and F. Pollara. Analysis, design, and iterative decoding of double serially concatenated codes with interlavers. *IEEE Journal on Selected Areas in Commun.*, 16(2):231–244, Feb. 1998.

[2] S. Benedetto and G. Montorsi. Unveiling turbo codes: Some results on parallel concatenated coding schemes. *IEEE Trans. Inform. Theory*, 42(2):409–428, March 1996.

[3] C. Berrou, A. Glavieux, and P. Thitimajshima. Near Shannon limit error-correcting coding and decoding: Turbo-codes. In *Proc. IEEE Int. Conf. Commun.*, volume 2, pages 1064–1070, Geneva, Switzerland, May 1993.

[4] D. Divsalar, H. Jin, and R. J. McEliece. Coding theorems for "turbo–like" codes. In *Proc. 36th Annual Allerton Conf. on Commun., Control, and Comp.*, pages 201–210, Monticello, IL, USA, Sept. 1998.

[5] T. Ericson. *Bounds on the Size of a Code*, pages 45–69. Number 128 in Lecture Notes in Control and Information Sciences. Springer-Verlag Berlin, Heidelberg, Germany, 1989.

[6] R. G. Gallager. Low-density parity-check codes. Research Monograph 21, The MIT Press, Cambridge, MA, 1963.

[7] R. G. Gallager. *Information Theory and Reliable Communications.* John Wiley and Sons, New York, NY, 1968.

[8] N. Kahale and R. Urbanke. On the minimum distance of parallel and serially concatenated codes. In *Proc. IEEE Int. Symposium on Inform. Theory*, page 31, Cambridge, MA, Aug. 1998.

[9] R. J. McEliece. Repeat accumulate codes [a class of turbo-like codes that we can analyse]. IMA Summer Program: Codes, Systems, and Graphical Models, Aug. 1999. http://www.ima.umn.edu/talks/workshops/aug2-13.99/mceliece/mceliece.html.

[10] M. Öberg and P. H. Siegel. Performance analysis of turbo-equalized dicode partial-response channel. In *Proc. 36th Annual Allerton Conf. on Commun., Control, and Comp.*, pages 230–239, Monticello, IL, USA, Sept. 1998.

[11] E. Seneta. *Non-Negative Matrices: An Introduction to Theory and Applications.* Wiley, New York, NY, 2nd edition, 1973.

Queue Length Asymptotics for Markovian Service Networks

A. Mandelbaum
Technion Institute
Haifa, 32000
ISRAEL
avim@tx.technion.ac.il

W. A. Massey
Bell Labs, Lucent Technologies
Murray Hill, NJ 07974
U.S.A.
will@research.bell-labs.com

M. I. Reiman
Bell Labs, Lucent Technologies
Murray Hill, NJ 07974
U.S.A.
marty@research.bell-labs.com

Abstract

We develop limit theorems for a large class of stochastic service network models that we call *Markovian service networks.* They are a special family of nonstationary Markov processes where parameters like arrival and service rates, routing topologies for the network, and the number of servers at a given node are all functions of time as well as the current state of the system. Using the theory of strong approximations we obtain both fluid and diffusion limit theorems for these processes. The asymptotic limiting regime that we explore for these networks has a natural interpretation of scaling up the number of servers in response to a similar scaling up of the arrival rate for the customers. The individual service rates however, are not scaled. Included in our modeling framework are Jackson networks of $M_t/M_t/n_t$ queues with abandonment and retrials. Both the stochastic models and the asymptotic scaling applied to them are inspired by service systems such as call centers.

1 Introduction

In papers [8] and [9], we facilitate the design and analysis of Markovian service networks, which we formally define in Section 2, by deriving fluid and diffusion limits for such systems. A recurring theme in most of the systems we consider is a large supply of servers and a corresponding large demand for this service. Our methods allow us to analyze networks with time dependent parameters, state dependent routing, abandonment, and retrials.

The asymptotic approach to the $M_t/M_t/1$ queue, as used in both Massey [13], [14] and Mandelbaum and Massey [7], was to create a family of associated $M_t/M_t/1$ queues where the queue indexed by $\eta > 0$ has arrival rate $\eta\lambda_t$ and service rate $\eta\mu_t$. We then determined the asymptotic behavior for the time evolution of this family of queues, when $\eta \to \infty$. Using the $M_t/M_t/n_t$ and the desire to model call centers as our guiding motivation, we generalize this type of analysis to all Markovian service networks.

In the case of call centers, service involves an interaction between either two people (the customer and server), or a person and a machine (the person is the customer). In either case, because a person is involved, it is not reasonable to scale the service rates with η. To accommodate the arrivals, whose rate is proportional to η, the number of servers must then be scaled with η. Time dependent arrival rates occur naturally, since phenomena such as rush hours are quite common. Time dependent service rates can be used to model phenomena such as server fatigue or changes in the nature of services over the day. Finally, a time dependent number of servers arises with shift changes and in systems where the number of servers is varied to accommodate changes in the arrival rate.

Our "first-order" asymptotic result takes the form of a *functional strong law of large numbers* (FSLLN), and yields a *fluid approximation* for the original process. Moreover, this fluid approximation is the solution to a dynamical system or an autonomous set of differential equations. In turn this FSLLN can be refined with a *functional central limit theorem* (FCLT). A fundamental difficulty arises in attempting to apply prior results (see Kurtz [5]) to obtain the FCLT. This difficulty is resolved for general Markovian service networks in [8]. To circumvent this difficulty, we introduce a new notion of derivative which we call the *scalable Lipschitz derivative*. Using this new notion of a "non-smooth" derivative we are able to obtain an FCLT for a wider class of stochastic models. From the FCLT we obtain a *diffusion approximation* which is a "second-order" asymptotic refinement of the fluid approximation. Similarly, both the mean vector and covariance tensor for the diffusion process are the unique solutions to dynamical systems. Finally, the SLLN and the FCLT both follow from a common *strong approximation theorem* (see Ethier and Kurtz [1]). The limit theorems then follow from a more detailed asymptotic analysis of this approximation theorem.

The precise specification of a Markovian service network and its requisite assumptions are presented in Section 2. The properties of scalable Lipschitz derivatives are stated in Section 3. Examples of service networks are presented in Sections 4 and 5. They include the construction of a Jackson network with multiserver nodes and abandonment. A special case of these networks include multiserver queues with abandonment and retrials for which we devote a paper [9] on the comparison of its fluid and diffusion approximations to a simulation of the actual queueing process. Another example that fits into the Markovian service network framework is a multiserver queue with preemptive priorities.

References to the large body of work on state dependent queues, time dependent queues, and related asymptotics can be found in [8] as well as Mandelbaum and Pats [11] and [12]. All the theorems stated in this paper are proved in [8]. Finally, we remark that we can extend this analysis of queue lengths to waiting times. The latter is the subject of a new paper [10].

2 The Model and Main Results

The primitives for our model are $\{ A_i(\cdot) \mid i \in I \}$, a collection of mutually independent, standard (rate 1) Poisson processes, indexed by a set I which is at most countably infinite; a separable Banach space $\mathbb{V}$ with norm $|\cdot|$; a collection of "jump" vectors $\{ \mathbf{v}_i \in \mathbb{V} \mid i \in I \}$ such that

$$\sum_{i\in I} |\mathbf{v}_i| < \infty; \tag{2.1}$$

a random initial state vector $\mathbf{Q}(0)$ in $\mathbb{V}$ that is always assumed to be independent of the collection of Poisson processes $\{ A_i(\cdot) \mid i \in I \}$; and a collection of real-valued, non-negative Lipschitz "rate" functions on $\mathbb{V}$,

$$\{ \alpha_t^\eta(\cdot, i) \mid t \geq 0,\, i \in I \}, \tag{2.2}$$

that jointly satisfy

$$\|\alpha_t^\eta(\cdot, i)\| \leq \eta \beta_t \gamma^{(i)} \tag{2.3}$$

for some β_t, a locally integrable function, and $\left\{ \gamma^{(i)} \mid i \in I \right\}$, a summable sequence of real numbers; here $\|\cdot\|$ is the *Lipschitz norm* for real-valued functions on $\mathbb{V}$, namely

$$\|f\| \equiv \sup_{\mathbf{x},\mathbf{y} \in \mathbb{V},\, \mathbf{x} \neq \mathbf{y}} \frac{|f(\mathbf{x}) - f(\mathbf{y})|}{|\mathbf{x} - \mathbf{y}|} \vee |f(0)| \tag{2.4}$$

and so f is a *Lipschitz function* whenever $\|f\| < \infty$.

For all of the examples that we consider in this paper, $\mathbb{V} = \mathbb{R}^N$ for some $1 \leq N < \infty$ and the number of elements in I is finite. Thus, although we prove the main results of the paper for a more general setting, any reader uncomfortable with the trappings of Banach spaces can replace $\mathbb{V}$ with $\mathbb{R}^N$ and still follow the examples we present. In that case $|\cdot|$ is the standard Euclidean norm on $\mathbb{R}^N$.

In terms of the primitives, we represent our *Markovian service network* to be the $\mathbb{V}$-valued stochastic process $\mathbf{Q}^\eta \equiv \{ \mathbf{Q}^\eta(t) \mid t \geq 0 \}$, whose sample paths are uniquely determined by $\mathbf{Q}^\eta(0)$ and the functional equations

$$\mathbf{Q}^\eta(t) = \mathbf{Q}^\eta(0) + \sum_{i \in I} A_i \left(\int_0^t \alpha_s^\eta\Big(\frac{1}{\eta}\mathbf{Q}^\eta(s), i\Big) ds \right) \mathbf{v}_i. \tag{2.5}$$

To construct the $M_t/M_t/n_t$ queue, for example, we let $\mathbb{V} = \mathbb{R}$, $I = \{1, 2\}$, $\mathbf{v}_1 = 1$, $\mathbf{v}_2 = -1$, $\alpha_t^\eta(x, 1) = \eta\lambda_t$, $\alpha_t^\eta(x, 2) = \mu_t \cdot (\eta x \wedge \eta n_t)$. Uniqueness of the solution to (2.5) is shown in [8]. Our goal is to characterize this asymptotic behavior as $\eta \uparrow \infty$ with a functional strong law of large numbers and a central limit theorem, but we do this with the following type of asymptotic behavior for the rate functions: for each $i \in I$, $\alpha_t^\eta(\cdot, i)$, as $\eta \to \infty$ we have

$$\alpha_t^\eta(\cdot, i) = \eta \alpha_t^{(0)}(\cdot, i) + \sqrt{\eta}\, \alpha_t^{(1)}(\cdot, i) + o(\sqrt{\eta}). \tag{2.6}$$

To define this asymptotic expansion rigorously, there are technical conditions that we spell out completely in [8].

Within the framework of strong approximations, we first approximate the sample-path representation (2.5) of the family $\{ \mathbf{Q}^\eta \mid \eta > 0 \}$ by the following theorem.

Theorem 2.1 (Strong Approximation) *As $\eta \to \infty$ we have*

$$\begin{aligned} \mathbf{Q}^\eta(t) &= \mathbf{Q}^\eta(0) + \int_0^t \boldsymbol{\alpha}_s^\eta\Big(\frac{1}{\eta}\mathbf{Q}^\eta(s)\Big) ds \\ &\quad + \sum_{i \in I} B_i \left(\int_0^t \alpha_s^\eta\Big(\frac{1}{\eta}\mathbf{Q}^\eta(s), i\Big) ds \right) \mathbf{v}_i + O(\log \eta) \quad \text{a.s.} \end{aligned} \tag{2.7}$$

where the convergence is uniform on compact sets in t, and

$$\boldsymbol{\alpha}_t^\eta(\mathbf{x}) = \sum_{i \in I} \alpha_t^\eta(\mathbf{x}, i) \mathbf{v}_i, \quad \mathbf{x} \in \mathbb{V}. \tag{2.8}$$

From this strong approximation, we deduce a FSLLN (Theorem 2.2), followed by a FCLT (Theorem 2.3). The limit theorems enable sample-path and distributional approximations, which support computations and confidence intervals.

Theorem 2.2 (FSLLN) *If* $\{\, \mathbf{Q}^{\eta}(0) \mid \eta > 0 \,\}$ *is any family of random initial state vectors in* $\mathbb{V}$*, then*

$$\lim_{\eta\to\infty} \frac{\mathbf{Q}^{\eta}(0)}{\eta} = \mathbf{Q}^{(0)}(0) \ \text{a.s.} \quad \text{implies} \quad \lim_{\eta\to\infty} \frac{\mathbf{Q}^{\eta}(t)}{\eta} = \mathbf{Q}^{(0)}(t) \ \text{a.s.} \tag{2.9}$$

where the convergence is uniform on compact sets in t*, and* $\mathbf{Q}^{(0)}$ *is the unique deterministic process* $\left\{\, \mathbf{Q}^{(0)}(t) \,\middle|\, t \geq 0 \,\right\}$ *that solves the integral equation*

$$\mathbf{Q}^{(0)}(t) = \mathbf{Q}^{(0)}(0) + \int_0^t \boldsymbol{\alpha}_s^{(0)}\Big(\mathbf{Q}^{(0)}(s)\Big)\, ds, \quad t \geq 0. \tag{2.10}$$

Here $\boldsymbol{\alpha}_t^{(0)}$*, given by*

$$\boldsymbol{\alpha}_t^{(0)}(\mathbf{x}) = \sum_{i\in I} \alpha_t^{(0)}(\mathbf{x}, i)\mathbf{v}_i, \quad \mathbf{x} \in \mathbb{V}, \tag{2.11}$$

is a Lipschitz mapping of $\mathbb{V}$ *into itself and its Lipschitz norm* $\left\|\boldsymbol{\alpha}_t^{(0)}\right\|$ *is a locally integrable function of* t*.*

In the development of a functional central limit theorem for our stochastic network, which refines the above fluid approximation, it is necessary to differentiate $\boldsymbol{\alpha}_t^{(0)}(\cdot)$ over the Banach space $\mathbb{V}$. There are specific examples of queueing systems that we analyze, like the $M_t/M_t/n_t$ queue, where the corresponding $\boldsymbol{\alpha}_t^{(0)}$ is piecewise differentiable but not everywhere differentiable. This poses a problem that is not easily ignored since these derivatives are evaluated at values for the fluid model $\mathbf{Q}^{(0)}(t)$. So even if $\boldsymbol{\alpha}_t^{(0)}(\cdot)$ has no derivative at only a finite number of points, the fluid process could spend all of its time at these points.

We resolve this issue by introducing a new type of differentiability. If $\mathbf{f}(\cdot)$ is a mapping from $\mathbb{V}_1$ into $\mathbb{V}_2$, we extend the Banach space norms $|\cdot|_1$ and $|\cdot|_2$ on $\mathbb{V}_1$ and $\mathbb{V}_2$ respectively, to define the following norm on $\mathbf{f}$:

$$\|\mathbf{f}\| \equiv \sup_{\mathbf{x},\mathbf{y}\in\mathbb{V}_1,\, \mathbf{x}\neq\mathbf{y}} \frac{|\mathbf{f}(\mathbf{x}) - \mathbf{f}(\mathbf{y})|_2}{|\mathbf{x} - \mathbf{y}|_1} \vee |\mathbf{f}(0)|_2, \tag{2.12}$$

and say that $\mathbf{f}$ is Lipschitz on $\mathbb{V}_1$ whenever $\|\mathbf{f}\| < \infty$.

Now we define $\mathbf{f}$ to have a *scalable Lipschitz derivative* at $\mathbf{x} \in \mathbb{V}_1$ if there exists another mapping from $\mathbb{V}_1$ into $\mathbb{V}_2$, denoted $\Lambda\mathbf{f}(\mathbf{x};\cdot)$, such that

$$\lim_{\mathbf{y}\to 0} \frac{|\mathbf{f}(\mathbf{x}+\mathbf{y}) - \mathbf{f}(\mathbf{x}) - \Lambda\mathbf{f}(\mathbf{x};\mathbf{y})|_2}{|\mathbf{y}|_1} = 0, \tag{2.13}$$

where the function $\Lambda\mathbf{f}(\mathbf{x};\cdot)$ is Lipschitz on $\mathbb{V}_1$ so that

$$\|\Lambda\mathbf{f}(\mathbf{x};\cdot)\| < \infty, \tag{2.14}$$

and for all real scalars with $\lambda \geq 0$,

$$\lambda\Lambda\mathbf{f}(\mathbf{x};\mathbf{y}) = \Lambda\mathbf{f}(\mathbf{x};\lambda\mathbf{y}). \tag{2.15}$$

Our construction of a non-smooth derivative can be viewed as the analogue to the multivariate definition of a derivative or constructing the Jacobian. We sometimes write this new derivative as

$$\Lambda \mathbf{f}_{\mathbf{x}}(\mathbf{y}) \equiv \Lambda \mathbf{f}(\mathbf{x};\mathbf{y}) \tag{2.16}$$

to emphasize that we should fix $\mathbf{x}$ and view the derivative as a function of $\mathbf{y}$.

We can now state the functional central limit theorem.

Theorem 2.3 (FCLT) *It follows that* $\boldsymbol{\alpha}_t^{(0)}$, *given by (2.11), and* $\boldsymbol{\alpha}_t^{(1)}$, *given by*

$$\boldsymbol{\alpha}_t^{(1)}(\mathbf{x}) = \sum_{i\in I} \alpha_t^{(1)}(\mathbf{x}, i)\mathbf{v}_i, \quad \mathbf{x} \in \mathbb{V}, \tag{2.17}$$

are both Lipschitz mappings of $\mathbb{V}$ *into itself, and their Lipschitz norms are locally integrable functions of* t,

Moreover, if we assume that $\boldsymbol{\alpha}_t^{(0)}(\cdot)$ *has a scalable Lipschitz derivative* $\Lambda\boldsymbol{\alpha}_t^{(0)}\big(\mathbf{Q}^{(0)}(t);\cdot\big)$ *and we have a family of random initial state vectors* $\{\, \mathbf{Q}^{\eta}(0) \mid \eta > 0 \,\}$ *in* $\mathbb{V}$, *then for all random vectors* $\mathbf{Q}^{(0)}(0)$ *and* $\mathbf{Q}^{(1)}(0)$ *in* $\mathbb{V}$, *it follows that*

$$\lim_{\eta\to\infty} \sqrt{\eta}\left[\frac{\mathbf{Q}^{\eta}(0)}{\eta} - \mathbf{Q}^{(0)}(0)\right] \overset{d}{=} \mathbf{Q}^{(1)}(0) \tag{2.18}$$

implies

$$\lim_{\eta\to\infty} \sqrt{\eta}\left[\frac{\mathbf{Q}^{\eta}(t)}{\eta} - \mathbf{Q}^{(0)}(t)\right] \overset{d}{=} \mathbf{Q}^{(1)}(t), \tag{2.19}$$

the convergence being weak-convergence in $D_{\mathbb{V}}[0,\infty)$, *the space of* $\mathbb{V}$*-valued functions that are right-continuous with left-limits, equipped with the Skorohod* J_1 *topology.*

Finally, the limit $\mathbf{Q}^{(1)} \equiv \big\{\, \mathbf{Q}^{(1)}(t) \,\big|\, t \geq 0 \,\big\}$ *is the unique stochastic process that solves the stochastic integral equation*

$$\begin{aligned}\mathbf{Q}^{(1)}(t) &= \mathbf{Q}^{(1)}(0) + \int_0^t \left[\Lambda\boldsymbol{\alpha}_s^{(0)}\big(\mathbf{Q}^{(0)}(s);\mathbf{Q}^{(1)}(s)\big) + \boldsymbol{\alpha}_s^{(1)}\big(\mathbf{Q}^{(0)}(s)\big)\right] ds \\ &\quad + \sum_{i\in I} B_i\left(\int_0^t \alpha_s^{(0)}\big(\mathbf{Q}^{(0)}(s), i\big) ds\right)\mathbf{v}_i, \quad t \geq 0,\end{aligned} \tag{2.20}$$

where the $\{\, B_i \mid i \in I \,\}$ *are a family of mutually independent, standard Brownian motions.*

One consequence of the diffusion limit is an associated set of differential equations that become useful in the computation of its mean and covariance matrix.

Theorem 2.4 *The mean vector and covariance matrix for* $\mathbf{Q}^{(1)}(t)$ *solve the following set of differential equations:*

$$\frac{d}{dt}\mathsf{E}\left[\mathbf{Q}^{(1)}(t)\right] = \mathsf{E}\left[\Lambda\boldsymbol{\alpha}_t^{(0)}\big(\mathbf{Q}^{(0)}(t);\mathbf{Q}^{(1)}(t)\big)\right] + \boldsymbol{\alpha}_t^{(1)}\big(\mathbf{Q}^{(0)}(t)\big). \tag{2.21}$$

and

$$\begin{aligned}\frac{d}{dt}\mathsf{Cov}\left[\mathbf{Q}^{(1)}(t), \mathbf{Q}^{(1)}(t)\right] &= \left\{\mathsf{Cov}\left[\mathbf{Q}^{(1)}(t), \Lambda\boldsymbol{\alpha}_t^{(0)}\big(\mathbf{Q}^{(0)}(t);\mathbf{Q}^{(1)}(t)\big)\right]\right\} \\ &\quad + \sum_{i\in I} \alpha_t^{(0)}\big(\mathbf{Q}^{(0)}(t), i\big)\mathbf{v}_i \otimes \mathbf{v}_i\end{aligned} \tag{2.22}$$

for almost all t, where

$$\mathsf{Cov}\left[\mathbf{Q}^{(1)}(t), \mathbf{Q}^{(1)}(t)\right] \equiv \mathsf{E}\left[\mathbf{Q}^{(1)}(t) \otimes \mathbf{Q}^{(1)}(t)\right] - \mathsf{E}\left[\mathbf{Q}^{(1)}(t)\right] \otimes \mathsf{E}\left[\mathbf{Q}^{(1)}(t)\right] \tag{2.23}$$

and for all vectors $\mathbf{x}$ *and* $\mathbf{y}$ *in* $\mathbb{V}$,

$$\{\mathbf{x} \otimes \mathbf{y}\} \equiv \mathbf{x} \otimes \mathbf{y} + \mathbf{y} \otimes \mathbf{x}. \tag{2.24}$$

Moreover, if $\Lambda\boldsymbol{\alpha}_t^{(0)}\left(\mathbf{Q}^{(0)}(t); \cdot\right)$ *is a linear operator for almost all t, then* $\mathsf{E}\left[\mathbf{Q}^{(1)}(t)\right]$ *is the unique solution for (2.21) and* $\mathsf{Cov}\left[\mathbf{Q}^{(1)}(t), \mathbf{Q}^{(1)}(t)\right]$ *is the unique solution for (2.22). Finally, for all* $s < t$, $\mathsf{Cov}\left[\mathbf{Q}^{(1)}(s), \mathbf{Q}^{(1)}(t)\right]$ *solves the same set of differential equations in t as does* $\mathsf{E}\left[\mathbf{Q}^{(1)}(t)\right]$*, but with a different set of initial conditions.*

3 Calculus for Scalable Lipschitz Derivatives

Certain basic properties of the scalable Lipschitz derivative are useful in doing calculations for the diffusion limits of our service network processes. We now state the general properties for these functions.

1. If $\mathbf{f} : \mathbb{V}_1 \to \mathbb{V}_2$ and $\mathbf{g} : \mathbb{V}_2 \to \mathbb{V}_3$ are both scalable Lipschitz differentiable at $\mathbf{f}(\mathbf{x})$, then $\mathbf{g} \circ \mathbf{f} : \mathbb{V}_1 \to \mathbb{V}_3$ is scalable Lipschitz differentiable at $\mathbf{x}$, with

$$\Lambda(\mathbf{g} \circ \mathbf{f})_{\mathbf{x}}(\mathbf{y}) = \left(\Lambda\mathbf{g}_{\mathbf{f}(\mathbf{x})} \circ \Lambda\mathbf{f}_{\mathbf{x}}\right)(\mathbf{y}). \tag{3.1}$$

2. If $\mathbf{f} : \mathbb{R}^n \to \mathbb{R}^m$ is differentiable at $\mathbf{x} \in \mathbb{R}^n$ with Jacobian matrix $D\mathbf{f}(\mathbf{x})$, then it is scalable Lipschitz differentiable there and its scalable Lipschitz derivative is matrix multiplication by the Jacobian matrix so that

$$\Lambda\mathbf{f}_{\mathbf{x}}(\mathbf{y}) = \mathbf{y} \cdot D\mathbf{f}(\mathbf{x}) \tag{3.2}$$

for all $\mathbf{y} \in \mathbb{R}^m$, viewing $\mathbf{y}$ as a row vector.

3. If $f : \mathbb{V} \to \mathbb{R}$ and $g : \mathbb{V} \to \mathbb{R}$ are both scalable Lipschitz differentiable at $\mathbf{x}$, then $f + g$, fg, and $f \vee g$ are all scalable Lipschitz differentiable at $\mathbf{x}$, where

$$\Lambda(\mathbf{f} + \mathbf{g})_{\mathbf{x}}(\mathbf{y}) = \Lambda\mathbf{f}_{\mathbf{x}}(\mathbf{y}) + \Lambda\mathbf{g}_{\mathbf{x}}(\mathbf{y}), \tag{3.3}$$

$$\Lambda(fg)_{\mathbf{x}}(\mathbf{y}) = f(\mathbf{x})\Lambda g_{\mathbf{x}}(\mathbf{y}) + g(\mathbf{x})\Lambda f_{\mathbf{x}}(\mathbf{y}). \tag{3.4}$$

and

$$\Lambda(f \vee g)_{\mathbf{x}}(\mathbf{y}) = 1_{\{f(\mathbf{x})>g(\mathbf{x})\}}\Lambda f_{\mathbf{x}}(\mathbf{y}) + 1_{\{f(\mathbf{x})<g(\mathbf{x})\}}\Lambda g_{\mathbf{x}}(\mathbf{y}) + 1_{\{f(\mathbf{x})=g(\mathbf{x})\}}\Lambda f_{\mathbf{x}}(\mathbf{y}) \vee \Lambda g_{\mathbf{x}}(\mathbf{y}). \tag{3.5}$$

One simple consequence of the second statement of this theorem is that if $f : \mathbb{R} \to \mathbb{R}$ has left and right derivatives everywhere, then it is everywhere scalable Lipschitz differentiable and

$$\Lambda f_x(y) = f'(x+)y^+ - f'(x-)y^-, \tag{3.6}$$

for all real x and y.

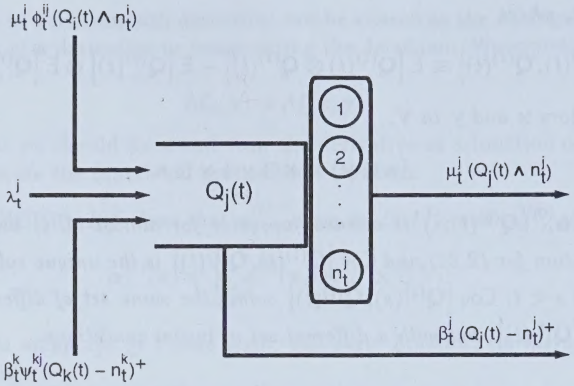

$\mu_t^i \phi_t^{ij}(Q_i(t) \wedge n_t^i)$
λ_t^j
$Q_j(t)$
1
2
n_t^j
$\mu_t^j (Q_j(t) \wedge n_t^j)$
$\beta_t^j (Q_j(t) - n_t^j)^+$
$\beta_t^k \psi_t^{kj} (Q_k(t) - n_t^k)^+$

Figure 1: The Jackson network with abandonment.

4 Jackson Networks with Abandonment

The multiserver queue with abandonment and retrials is a special case of a more general network that we discuss in this section. Here, we construct a time varying analogue of the Jackson network that has the added feature of service abandonment (see Figure 1). We construct it by first defining the following set of parameters:

$\lambda_t^i =$ external arrival rate to node i at time t,
$\beta_t^i =$ abandonment rate for node i at time t,
$\mu_t^i =$ service rate for node i at time t,
$\psi_t^{ij} =$ abandonment routing probability from node i to node j at time t,
$\phi_t^{ij} =$ service routing probability from node i to node j at time t,
$\psi_t^i =$ abandonment departure probability from node i at time t,
$\phi_t^i =$ service departure probability from node i at time t.
$n_t^i =$ number of servers for node i at time t,

where we require that

$$\psi_t^i + \sum_{j=1}^{N} \psi_t^{ij} = 1 \quad \text{and} \quad \phi_t^i + \sum_{j=1}^{N} \phi_t^{ij} = 1 \tag{4.1}$$

for all $t \geq 0$ and $i = 1, \ldots, N$.

We then set $\mathbb{V} = \mathbb{R}^N$ and define

$$\begin{aligned} \mathbf{Q}(t) \;&=\; \mathbf{Q}(0) + \sum_{i=1}^{N}\sum_{j=1}^{N} A_{ij}^{b} \left(\int_0^t \left(Q_i(s) - n_s^i \right)^+ \beta_s^i \psi_s^{ij} ds \right) (\mathbf{e}_j - \mathbf{e}_i) \\ &\quad + \sum_{i=1}^{N}\sum_{j=1}^{N} A_{ij}^{c} \left(\int_0^t \left(Q_i(s) \wedge n_s^i \right) \mu_s^i \phi_s^{ij} ds \right) (\mathbf{e}_j - \mathbf{e}_i) \end{aligned} \tag{4.2}$$

$$
\begin{aligned}
&+\sum_{i=1}^{N}\left(A_i^a\left(\int_0^t \lambda_s^i ds\right) - A_i^b\left(\int_0^t \left(Q_i(s) - n_s^i\right)^+ \beta_s^i \psi_s^i ds\right)\right) \mathbf{e}_i \\
&-\sum_{i=1}^{N} A_i^c\left(\int_0^t \left(Q_i(s) \wedge n_s^i\right) \mu_s^i \phi_s^i ds\right) \mathbf{e}_i .
\end{aligned}
$$

Theorems 2.2 and 2.3 yield the following limiting results for these networks.

Theorem 4.1 *The fluid limit for the Jackson network with abandonment is the unique solution to the integral equation*

$$
\begin{aligned}
\mathbf{Q}^{(0)}(t) &= \mathbf{Q}^{(0)}(0) + \int_0^t \left[\boldsymbol{\lambda}_s + (\mathbf{Q}^{(0)}(s) - \mathbf{n}_s)^+ \boldsymbol{\Delta}(\boldsymbol{\beta}_s)(\Psi_s - \mathbf{I})\right] ds \qquad (4.3) \\
&\quad + \int_0^t (\mathbf{Q}^{(0)}(s) \wedge \mathbf{n}_s)\boldsymbol{\Delta}(\boldsymbol{\mu}_s)(\Phi_s - \mathbf{I}) ds.
\end{aligned}
$$

Theorem 4.2 *The mean vector for the diffusion limit solves the differential equation*

$$
\begin{aligned}
\frac{d}{dt}\mathsf{E}\left[\mathbf{Q}^{(1)}(t)\right] &= \mathsf{E}\left[\mathbf{Q}^{(1)}(t)^+ \mathbf{I}_{\{\mathbf{Q}^{(0)}(t) \geq \mathbf{n}_t\}} - \mathbf{Q}^{(1)}(t)^- \mathbf{I}_{\{\mathbf{Q}^{(0)}(t) > \mathbf{n}_t\}}\right] \boldsymbol{\Delta}(\boldsymbol{\beta}_t)(\Psi_t - \mathbf{I}) \\
&\quad + \mathsf{E}\left[\mathbf{Q}^{(1)}(t)^+ \mathbf{I}_{\{\mathbf{Q}^{(0)}(t) < \mathbf{n}_t\}} - \mathbf{Q}^{(1)}(t)^- \mathbf{I}_{\{\mathbf{Q}^{(0)}(t) \leq \mathbf{n}_t\}}\right] \boldsymbol{\Delta}(\boldsymbol{\mu}_t)(\Phi_t - \mathbf{I}),
\end{aligned}
$$

and the covariance matrix for the diffusion limit solves the differential equation

$$
\begin{aligned}
&\frac{d}{dt}\mathsf{Cov}\left[\mathbf{Q}^{(1)}(t), \mathbf{Q}^{(1)}(t)\right] \\
&= \left\{\mathsf{Cov}\left[\mathbf{Q}^{(1)}(t), \mathbf{Q}^{(1)}(t)^+ \mathbf{I}_{\{\mathbf{Q}^{(0)}(t) \geq \mathbf{n}_t\}} - \mathbf{Q}^{(1)}(t)^- \mathbf{I}_{\{\mathbf{Q}^{(0)}(t) > \mathbf{n}_t\}}\right] \boldsymbol{\Delta}(\boldsymbol{\beta}_t)(\Psi_t - \mathbf{I})\right\} \\
&\quad + \left\{\mathsf{Cov}\left[\mathbf{Q}^{(1)}(t), \mathbf{Q}^{(1)}(t)^+ \mathbf{I}_{\{\mathbf{Q}^{(0)}(t) < \mathbf{n}_t\}} - \mathbf{Q}^{(1)}(t)^- \mathbf{I}_{\{\mathbf{Q}^{(0)}(t) \leq \mathbf{n}_t\}}\right] \boldsymbol{\Delta}(\boldsymbol{\mu}_t)(\Phi_t - \mathbf{I})\right\} \\
&\quad + \boldsymbol{\Delta}\left(\boldsymbol{\lambda}_t + \left(Q^{(0)}(t) - \mathbf{n}_t\right)^+ \boldsymbol{\Delta}(\boldsymbol{\beta}_t)(\Psi_t + \mathbf{I}) + \left(Q^{(0)}(t) \wedge \mathbf{n}_t\right)\boldsymbol{\Delta}(\boldsymbol{\mu}_t)(\Phi_t + \mathbf{I})\right) \\
&\quad - \left\{\boldsymbol{\Delta}\left((Q^{(0)}(t) - \mathbf{n}_t)^+\right)\boldsymbol{\Delta}(\boldsymbol{\beta}_t)\Psi_t + \boldsymbol{\Delta}\left(Q^{(0)}(t) \wedge \mathbf{n}_t\right)\boldsymbol{\Delta}(\boldsymbol{\mu}_t)\Phi_t\right\}.
\end{aligned}
$$

5 Priority Queues

A multiserver queue with preemptive priorities (see Figure 2) is defined using the following parameters:

$\lambda_t^i =$ arrival rate for class i customers at time t,
$\mu_t^i =$ service rate for class i customers at time t,
$n_t =$ number of servers at time t,
$c =$ number of customer classes.

We then set $\mathbb{V} = \mathbb{R}^c$ and define

$$
\mathbf{Q}(t) = \mathbf{Q}(0) + \sum_{i=1}^{c}\left[A_i^a\left(\int_0^t \lambda_s^i ds\right) - A_i^b\left(\int_0^t \mu_s^i\, Q_i(s) \wedge \left(n_t - \sum_{j=1}^{i-1} Q_j(s)\right)^+ ds\right)\right] \mathbf{e}_i. \qquad (5.1)
$$

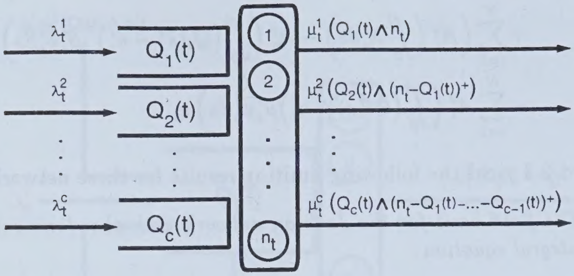

λ¹_t
Q₁(t)
1
μ¹_t (Q₁(t)∧n_t)
λ²_t
2
μ²_t (Q₂(t)∧(n_t−Q₁(t))⁺)
Q₂(t)
λᶜ_t
μᶜ_t (Q_c(t)∧(n_t−Q₁(t)−...−Q_{c−1}(t))⁺)
Q_c(t)
n_t

Figure 2: The preemptive priority queue.

Theorem 5.1 *The fluid limit for the priority queueing model is the solution to the integral equation*

$$\mathbf{Q}^{(0)}(t) = \mathbf{Q}^{(0)}(0) + \int_0^t \left[\boldsymbol{\lambda}_s + \left(\mathbf{Q}^{(0)}(s) \wedge (n_s\mathbf{1} - \mathbf{Q}^{(0)}(s)\boldsymbol{\Theta})\right)^+ \boldsymbol{\Delta}(\boldsymbol{\mu}_s)\right] ds, \tag{5.2}$$

where $\boldsymbol{\Theta} = \{\, \theta_{ij} \mid 1 \le i, j \le c\}$ *is the* $c \times c$ *matrix*

$$\theta_{ij} = \begin{cases} 1 & \text{for } i < j, \\ 0 & \text{for } i \ge j. \end{cases} \tag{5.3}$$

Theorem 5.2 *The mean vector for the diffusion limit solves the differential equation*

$$\begin{aligned} \frac{d}{dt}\mathsf{E}\left[\mathbf{Q}^{(1)}(t)\right] &= \mathsf{E}\left[\mathbf{Q}^{(1)}(t)\right] \mathbf{I}_{\{\mathbf{Q}^{(0)}(t)<(n_t\mathbf{1}-\mathbf{x}\boldsymbol{\Theta})^+\}} \\ &\quad + \mathsf{E}\left[\left(\mathbf{Q}^{(1)}(t)\boldsymbol{\Theta}\right)^+\right] \mathbf{I}_{\{\mathbf{Q}^{(0)}(t)>(n_t\mathbf{1}-\mathbf{Q}^{(0)}(t)\boldsymbol{\Theta})^+,\ n_t\mathbf{1}\le\mathbf{Q}^{(0)}(t)\boldsymbol{\Theta}\}} \\ &\quad - \mathsf{E}\left[\left(\mathbf{Q}^{(1)}(t)\boldsymbol{\Theta}\right)^-\right] \mathbf{I}_{\{\mathbf{Q}^{(0)}(t)>(n_t\mathbf{1}-\mathbf{Q}^{(0)}(t)\boldsymbol{\Theta})^+,\ n_t\mathbf{1}<\mathbf{Q}^{(0)}(t)\boldsymbol{\Theta}\}} \\ &\quad + \mathsf{E}\left[\mathbf{Q}^{(1)}(t) \wedge \left(-(\mathbf{Q}^{(1)}(t)\boldsymbol{\Theta})^*\right)\right] \cdot \mathbf{I}_{\{\mathbf{Q}^{(0)}(t)=(n_t\mathbf{1}-\mathbf{Q}^{(0)}(t)\boldsymbol{\Theta})^+\}} \end{aligned}$$

and the covariance matrix for the diffusion limit solves the differential equation

$$\begin{aligned} &\frac{d}{dt}\mathsf{Cov}\left[\mathbf{Q}^{(1)}(t), \mathbf{Q}^{(1)}(t)\right] \\ &\quad = \left\{\mathsf{Cov}\left[\mathbf{Q}^{(1)}(t), \mathbf{Q}^{(1)}(t)\right] \mathbf{I}_{\{\mathbf{Q}^{(0)}(t)<(n_t\mathbf{1}-\mathbf{x}\boldsymbol{\Theta})^+\}}\right\} \\ &\qquad + \left\{\mathsf{Cov}\left[\mathbf{Q}^{(1)}(t), \left(\mathbf{Q}^{(1)}(t)\boldsymbol{\Theta}\right)^+\right] \mathbf{I}_{\{\mathbf{Q}^{(0)}(t)>(n_t\mathbf{1}-\mathbf{Q}^{(0)}(t)\boldsymbol{\Theta})^+,\ n_t\mathbf{1}\le\mathbf{Q}^{(0)}(t)\boldsymbol{\Theta}\}}\right\} \\ &\qquad - \left\{\mathsf{Cov}\left[\mathbf{Q}^{(1)}(t), \left(\mathbf{Q}^{(1)}(t)\boldsymbol{\Theta}\right)^-\right] \mathbf{I}_{\{\mathbf{Q}^{(0)}(t)>(n_t\mathbf{1}-\mathbf{Q}^{(0)}(t)\boldsymbol{\Theta})^+,\ n_t\mathbf{1}<\mathbf{Q}^{(0)}(t)\boldsymbol{\Theta}\}}\right\} \\ &\qquad + \left\{\mathsf{Cov}\left[\mathbf{Q}^{(1)}(t), \mathbf{Q}^{(1)}(t) \cdot\wedge \left(-(\mathbf{Q}^{(1)}(t)\boldsymbol{\Theta})^*\right)\right] \cdot \mathbf{I}_{\{\mathbf{Q}^{(0)}(t)=(n_t\mathbf{1}-\mathbf{Q}^{(0)}(t)\boldsymbol{\Theta})^+\}}\right\} \\ &\qquad + \boldsymbol{\Delta}(\boldsymbol{\lambda}_t) - \boldsymbol{\Delta}\left(\mathbf{Q}^{(0)}(t) \wedge \left(n_t\mathbf{1} - \mathbf{Q}^{(0)}(t)\boldsymbol{\Theta}\right)^+\right) \cdot \boldsymbol{\Delta}(\boldsymbol{\mu}_t). \end{aligned}$$

where

$$(\mathbf{Q}^{(1)}(t)\boldsymbol{\Theta})^* = (\mathbf{Q}^{(1)}(t)\boldsymbol{\Theta})^+\mathbf{I}_{\{n_t\mathbf{1}<\mathbf{Q}^{(0)}(t)\boldsymbol{\Theta}\}} - (\mathbf{Q}^{(1)}(t)\boldsymbol{\Theta})^-\mathbf{I}_{\{n_t\mathbf{1}\le\mathbf{Q}^{(0)}(t)\boldsymbol{\Theta}\}}. \tag{5.4}$$

References

[1] S. N. Ethier and T. G. Kurtz. *Markov Process: Characterization and Convergence.* John Wiley and Sons, New York, 1986.

[2] S. Halfin and W. Whitt. Heavy-traffic limits theorem for queues with many exponential servers. *Operations Research*, 29:567–588, 1981.

[3] D. L. Iglehart. Limit diffusion approximations for the many server queue and the repairman problem. *Journal of Applied Probability*, 2:429–441, 1965.

[4] J.R. Jackson. Jobshop-like queueing systems. *Management Science*, 10(1):131–142, 1963.

[5] T. G. Kurtz. Strong approximation theorems for density dependent Markov chains. *Stochastic Processes and Their Applications*, 6:223–240, 1978.

[6] T. G. Kurtz. Representation of Markov processes as multiparameter time changes. *The Annals of Probability*, 8(4):682–715, 1980.

[7] A. Mandelbaum and W. A. Massey. Strong approximations for time-dependent queues. *Mathematics of Operations Research*, 20(1):33–64, 1995.

[8] A. Mandelbaum, W. A. Massey, M. I. Reiman. Strong Approximations for Markovian Service Networks. *Queueing Systems*, (1998).

[9] A. Mandelbaum, W. A. Massey, M. I. Reiman, B. Rider. Time Varying Multiserver Queues with Abandonment and Retrials. *ITC-16*, Edinburgh, Scotland, (1999).

[10] A. Mandelbaum, W. A. Massey, M. I. Reiman, A. L. Stolyar. Waiting Time Asymptotics for Multiserver, Nonstationary Jackson Networks with Abandonment. *In preparation.*

[11] A. Mandelbaum and G. Pats. State-dependent queues: Approximation and applications. In F. Kelly and R.J. Williams, editors, *IMA Volumes in Mathematics and Its Applications*, volume 71, pages 239–282. Springer-Verlag, 1995.

[12] A. Mandelbaum and G. Pats. State-dependent stochastic networks, Part I: Approximations and applications with continuous diffusion limits. *Annals of Applied Probability*, 8(2):569-646, 1998.

[13] W.A. Massey. Asymptotic analysis of the time dependent M/M/1 queue, *Math. of Op. Res.*, **10**, 305–327, 1985.

[14] W.A. Massey. *Nonstationary Queues*, Ph. D. Thesis, Stanford University, 1981.

[15] G. Pats. *State-Dependent Queueing Networks: Approximations and Applications.* PhD thesis, Faculty of Industrial Engineering and Management, Technion, 1994.

Exact Asymptotics for 1-Limited Exponential Polling Models

Woojin Chang, Douglas G. Down*and Robert D. Foley
School of Industrial and Systems Engineering
Georgia Institute of Technology
Atlanta, GA 30332-0205

Abstract

The exact asymptotics for rare events in polling models is an area that has seen little study. In this paper we find exact asymptotic expressions for the event that the total queue length is large in a 2 customer, 1-limited exponential polling model with equal service rates. It is found that this behavior divides into two very different regimes, depending on the arrival rates to the system.

1 Introduction

Polling models have seen wide application in modelling systems in the diverse areas of telecommunications, transportation, computer performance, inventory, etc. (for a couple of general references, see Boxma and Takagi [1] and Levy and Sidi [6]). A (single server) polling model is simple to describe: there are several classes of customers arriving to a single server which operates under a particular service policy. Some typical examples are the so-called *exhaustive* policy, in which the server empties the system of a class of customers before moving on to the next class and *limited* policies, in which the server has a limit on the number of customers of a class that can be served before moving on to the next class. The latter is the focus of this paper.

Over the last several years, there has been a flurry of activity in the analysis of rare events (or large deviations) in stochastic systems. It has been argued that in many practical systems, a useful performance measure is the frequency of rare events, such as data loss in a telecommunications system or large delays in filling an order at a warehouse. Thus it is somewhat surprising that there is a dearth of results for rare events in polling models. The authors are aware of only two papers addressing this problem (Choudhury and Whitt [3] and Duffield [4]) and these deal specifically with polling models under exhaustive-type service policies. The first of these papers uses numerical inversion techniques on the moment generating function for the invariant distribution of the system while the latter uses more "traditional" large deviations techniques. We are not aware of any articles that analyze rare events for polling models with limited service policies. In fact, [3] states that limited service policies do not fit certain structural assumptions that allows their analysis to proceed. As the same structural assumptions are employed in [4], it appears that these techniques will have difficulty seeing application to limited service policies.

---

*Corresponding author. email: ddown@isye.gatech.edu

Recently, McDonald [7] has developed a methodology for analyzing rare events for Markov chains evolving on an orthant. We have found that the analysis of polling models with limited service policies is tailor-made for this methodology. The crux of this methodology is that engineering insight must be employed to identify possible large deviations paths a priori and then verify some technical conditions. We have found that it is not difficult to identify these paths and that the required technical conditions tend to be fairly straightforward to verify. In this paper, we hope to give an indication of what kind of results may be achieved by examining the simplest possible system, one with two classes of customers served under the 1-limited service policy, where the service requirements for both classes of customers are the same. This model will be made more precise in the next section. Results for more general limited service policies and for unequal service requirements are currently in preparation. We would like to comment at this point that these generalizations are not conceptually any more difficult than the case studied in this paper, but do become somewhat more difficult from a technical standpoint.

The layout of the paper is as follows. In Section 2, we present the model and the appropriate Markov chain that describes its behavior. Section 3 gives the main results, while Section 4 discusses McDonald's methodology and its application to our system (thus providing the justification of the results in the previous section). Finally, Section 5 provides some concluding remarks.

For the most part, proofs have been omitted throughout the paper. There is a companion paper [2] which will contain these proofs as well as an extension of these results to general k_i-limited policies, where the number served at each visit to a class may be class dependent (and greater than one).

2 The Model

We consider a system with 2 distinct classes of customers. Customers of class 1 and class 2 arrive to the system according to independent Poisson processes with rates λ_1 and λ_2, respectively. Without loss of generality, we will assume that the classes have been labelled such that $\lambda_2 > \lambda_1$. There is one server and the service times for a customer (regardless of class) are exponentially distributed with rate μ. The service time for a customer is independent of all other service times as well as the arrival processes. Service is performed in a *1-limited* manner. That is, nominally when the server is visiting class i, it serves exactly one customer of class i (if there is one available) before visiting the other class. The service policy is completely specified upon noting that it is non-idling, i.e. if there are no customers of class 1 (class 2) in the system, the server continues to serve class 2 (class 1) until it can return to nominal operation. Service is non-preemptive.

It is simple to verify that $Q(t) = (Q_1(t), Q_2(t), Z(t))$ is a continuous time Markov chain, where $Q_i(t)$ is the number of customers of class i in the system at time t and $Z(t)$ is the class that is currently being served by the system. We arbitrarily set $Z(t) = 2$ if no customers are in the system. The Markov chain $Q(t)$ thus evolves on $S = Z_+ \times Z_+ \times \{1, 2\}$. For the rest of this paper, we will look at the equivalent *uniformized* chain, $Q[n]$, where we assume without loss of generality that we have rescaled time such that $\lambda_1 + \lambda_2 + 2\mu = 1$. Let the transition kernel for this chain be denoted by K, where $K(x, y)$ gives the probability of going to state y, given the chain is currently in state x. For example, if

$x = (i, j, 2)$, with $i, j > 0$,

$$\begin{aligned} K(x, (i+1, j, 2)) &= \lambda_1 \\ K(x, (i, j+1, 2)) &= \lambda_2 \\ K(x, (i, j-1, 1)) &= \mu \\ K(x, (i, j, 2)) &= \mu. \end{aligned}$$

The existence of an invariant distribution π_Q is guaranteed if $\rho := (\lambda_1 + \lambda_2)/\mu < 1$. From this point on, we shall assume that this stability condition holds.

The event that we will be interested in in this paper is that the total system size is ℓ, i.e. the performance measure of interest is $F_\ell = \mathrm{P}\{Q_1[n] + Q_2[n] = \ell\}$. The procedure is similar if we were interested in related events such as the probability that one particular queue length is large.

3 Main Results

We will find that there are two distinct types of behavior, which depend on the following condition

$$\frac{2\lambda_1 \mu}{(\lambda_1 + \lambda_2)^2}. \tag{1}$$

If (1) is less than 1, we call the behavior *single-class dominant*, for reasons that we will later see. If (1) is greater than or equal to 1, we will call the behavior *non single-class dominant.*

We say $f \sim g$ if $f(\ell)/g(\ell) \to 1$ as $\ell \to \infty$. Let T_ℓ denote the first time that there are ℓ or more customers in the system.

Theorem 1 *For a single-class dominant system,*
(i)

$$\mathrm{E}[T_\ell | Q(0) = (0, 0, 2)] \sim g^{-1} \rho^{-\ell}$$

where the constant g may be obtained by a fast simulation and is given in (6) below.
(ii) For non-negative integers ℓ and j, and $k \in \{1, 2\}$,

$$\mathrm{P}\{Q_2(t) = \ell - j, Q_1(t) = j, Z(t) = k\} \sim \frac{g}{\mu - (\lambda_1 + \lambda_2)} \rho^\ell \varphi(j, k) \tag{2}$$

where φ is given in Lemma 1 below.

Theorem 2 *For a non single-class dominant system*
(i)

$$\mathrm{E}[T_\ell | Q(0) = (0, 0, 2)] \sim g^{-1} \rho^{-\ell}$$

where the constant g may be obtained by a fast simulation and is given in (8) below.
(ii) For any non-negative integer ℓ and $k \in \{1, 2\}$,

$$\mathrm{P}\{Q_1(t) + Q_2(t) = \ell, Z(t) = k\} \sim \frac{g}{2(\mu - (\lambda_1 + \lambda_2))} \rho^\ell. \tag{3}$$

(iii)

$$\lim_{\ell \to \infty} \left(\frac{Q_1(T_\ell)}{\ell}, \frac{Q_2(T_\ell)}{\ell} \right) \longrightarrow \left(\frac{\lambda_1 \rho^{-1} - \frac{\lambda_1 + \lambda_2}{2}}{\mu - (\lambda_1 + \lambda_2)}, \frac{\lambda_2 \rho^{-1} - \frac{\lambda_1 + \lambda_2}{2}}{\mu - (\lambda_1 + \lambda_2)} \right) \tag{4}$$

At this point, several remarks are in order. We have provided exact asymptotics for the 1-limited system (they are not quite closed form, as g may be obtained by a fast simulation). The fact that the rate in (2) and (3) is simply ρ^ℓ should not be surprising, as $Q_1(t) + Q_2(t)$ acts like the queue length in an M/M/1 system with arrival rate $\lambda_1 + \lambda_2$ and service rate μ.

Given that the rate is easily calculated, the usefulness of our results is twofold. First, we have identified the region of the parameter space for which the two different types of asymptotic behavior occur. For the single-class dominant system, looking at (2), we can see that $Q_1(t) + Q_2(t)$ reaches some large value ℓ with $Q_1(t)$ staying "close" to zero. Of course, we have gone much further than this by explicitly giving the distributions of the two queue lengths and the position of the server when this large level has been hit. This latter explicit solution is the second useful feature.

For the non single-class dominant system, we see that on the approach to F_ℓ both queue lengths get "large". Thus the asymptotics are essentially those of an M/M/1 queue, with the additional information contained in (4) which indicates where we expect to hit the line $Q_1(t) + Q_2(t) = \ell$.

We would expect these results to be useful for such problems as buffer allocation. Unfortunately, we have yet to find a physical interpretation for the condition (1) in terms of the original network. Rather, the interpretation is in terms of one of the constructed chains ($\mathcal{W}^\infty$). Such insight could be useful for system design.

4 Justification of Main Results

In the interests of space, we have chosen not to go into detail in describing the methodology of McDonald [7]. There is an excellent summary of the methodology given in Foley and McDonald [5], where it is applied to a particular Join the Shortest Queue system. It would probably be useful to the reader to have [5] in front of herself when reading this section.

4.1 Single-class dominant (Theorem 1)

Recall that we defined the system to be class 2 dominant if $2\lambda_1\mu(\lambda_1 + \lambda_2)^{-2} < 1$. We expect F_ℓ to be reached only through the impact of class 2 customers. To this end, we first define a chain W such that $W[n] = (\tilde{W}[n], \hat{W}[n])$ such that the components of $\tilde{W}[n]$ diverge as ℓ gets large and $\tilde{W}_1[n] \geq \ell$ when the event F_ℓ occurs. In this case, we simply choose $\tilde{W}[n]$ to be $Q_1[n] + Q_2[n]$ and $\hat{W}[n]$ to be the ordered pair $(Q_1[n], Z[n])$. The transition kernel for this new chain is denoted K_W. An example of the structure of K_W is, if $x = (i, 0, 2)$, with $i > 0$,

$$\begin{aligned}
K_W(x, (i+1, 1, 2)) &= \lambda_1 \\
K_W(x, (i+1, 0, 2)) &= \lambda_2 \\
K_W(x, (i, 0, 2)) &= \mu \\
K_W(x, (i-1, 0, 2)) &= \mu.
\end{aligned}$$

Next, we construct the "free" chain (W^∞) which simply means removing the boundaries corresponding to those components in $\tilde{W}$ and forcing W^∞ to have an additive structure. In our example, this boundary is denoted $\triangle$ and is all states of the form $(0, 0, 2)$ and $(x, x, 1), x \in Z_+$. Perhaps this change is best described for our example by saying that W^∞ is constructed from W by removing the boundary that queue 2 must be nonnegative, i.e. the server *always* serves one customer at queue 2 and a negative number of customers is allowed. After doing an appropriate change of measure, we will find that with positive probability that expanding the state space in this manner does not change the state trajectory. In this case, we let K^∞ be the transition kernel for W^∞ so for example, if $x = (i, i, 1)$,

$$\begin{aligned}
K^\infty(x, (i+1, i+1, 1)) &= \lambda_1 \\
K^\infty(x, (i+1, i, 1)) &= \lambda_2 \\
K^\infty(x, (i-1, i-1, 2)) &= \mu \\
K^\infty(x, (i, i, 2)) &= \mu.
\end{aligned}$$

Note that the last term would have been $K^\infty(x, (i, i, 1)) = \mu$ for the original system. The chain W^∞ evolves on the expanded state space $Z \times Z_+ \times \{1, 2\}$ and it is straightforward to show that it has the following additive structure

$$\begin{aligned}
&K^\infty((i, (j, \gamma)), (k, (l, \beta))) \\
&= \mathrm{P}\{\tilde{W}^\infty[n+1] = k, \hat{W}^\infty[n+1] = (l, \beta) | \tilde{W}^\infty[n+1] = i, \hat{W}^\infty[n+1] = (j, \gamma)\}.
\end{aligned}$$

Next, we have to construct the *twisted free* ($\mathcal{W}^\infty$) chain. The idea here is that we wish to twist the underlying distributions such that our choice of the most likely path to F_ℓ is emphasized. To do this we need to find a harmonic function for the free process of the form $h(\tilde{x}, \hat{x}) = \alpha^{\tilde{x}_1}\hat{a}(\hat{x})$, where $\tilde{x}$ corresponds to $\tilde{W}^\infty$, $\hat{x}$ corresponds to $\hat{W}^\infty$ and $\tilde{x}_1$ is the first component of $\tilde{x}$. In this case, it is straightforward to verify that $\alpha = \rho^{-1}$ and $\hat{a}(\hat{x}) \equiv 1$ is an appropriate harmonic function by verifying that $\mathrm{E}[h(W^\infty[n+1]) | W^\infty[n]] = h(W^\infty[n])$. The transition kernel for $\mathcal{W}^\infty$, $\mathcal{K}^\infty$, is constructed by twisting K^∞ as follows:

$$\mathcal{K}^\infty(x, y) = K^\infty(x, y)\frac{h(x)}{h(y)}.$$

Finally, we need to show that technical Conditions 0-7 in [5] hold.

Condition 0. Let $\tilde{d}_1 = \mathrm{E}[\tilde{\mathcal{W}}^\infty[n+1]] - \mathrm{E}[\tilde{\mathcal{W}}^\infty[n]]$. We must verify $\tilde{d}_1 > 0$, or in other words that the twisted free process drifts towards F_ℓ. In our case, $\tilde{d}_1 = \mu - (\lambda_1 + \lambda_2)$ which is positive by assumption.

Condition 1. Here, we must show that $\hat{\mathcal{W}}^\infty$ is irreducible and has an invariant distribution φ. Irreducibility is obvious, so we will now sketch why φ exists. In preparation to doing this, let $\tilde{\lambda}_i = \lambda_i/\rho$ and $\tilde{\mu} = \mu\rho$. By sampling the chain at times that the server returns to queue 1, we have that when there is a positive number of class 1 customers in the system, for $\mathcal{W}^\infty$

$$\begin{aligned}
\mathrm{E}[\text{time between visits to class 1}] &= \frac{2}{\tilde{\mu}} \\
\mathrm{E}[\text{number of arrivals of class 1 between visits}] &= \frac{2}{\tilde{\mu}}\tilde{\lambda}_1.
\end{aligned}$$

Now exactly one class 1 customer is served in this time period, so we have that φ exists iff

$$\frac{2}{\tilde{\mu}}\tilde{\lambda}_1 < 1$$

which is exactly the single-class dominance condition.

In fact, we can proceed further and explicitly calculate φ. With some algebra, the following lemma can be verified by direct substitution into the balance equations for φ along with the normalizing condition.

Lemma 1 *The invariant distribution $\varphi(j,k)$ is given by*

$$\begin{aligned}\varphi(j,1) &= \frac{\tilde{\mu}(1-x)}{\tilde{\lambda}_1+\tilde{\mu}(1+x)}x^j \qquad j \geq 1 \\ \varphi(j,2) &= \frac{(\tilde{\lambda}_1+\tilde{\mu})(1-x)}{\tilde{\lambda}_1+\tilde{\mu}(1+x)}x^j \qquad j \geq 1\end{aligned}$$

where

$$x = \frac{(\tilde{\lambda}_1+\tilde{\mu})^2 - \sqrt{(\tilde{\lambda}_1+\tilde{\mu})^4 - 4\tilde{\lambda}_1\tilde{\mu}^2(\tilde{\lambda}_1+\tilde{\mu})}}{2\tilde{\mu}^2}.$$

It is not obvious that $0 < x < 1$ as is required for φ to be a valid distribution. The demonstration of this also requires some algebra, but has been omitted due to space considerations. However it is important to note that the class 2 dominance condition implies this result.

Condition 2. The fact that $\mathcal{W}^\infty$ is aperiodic follows from the fact that there is a transition from state x to state x for all x.

Condition 3. This is subsumed by Condition 0.

Condition 4. Here we wish to verify that with positive probability $\mathcal{W}^\infty$ wanders away from Δ without returning to Δ. This follows from the fact that $\tilde{d}_1$ is positive and $\hat{\mathcal{W}}^\infty$ drifts towards 0.

Condition 5. Verify $\sum_{\hat{z}\in Z_+\times\{1,2\}} \varphi(\hat{z})/\hat{a}(\hat{z}) < \infty$. This is trivial as $\hat{a}(\cdot) \equiv 1$.

Condition 6. We need to verify

$$\lambda(x) = \sum_{z\in\Delta} \pi_W(z)K_W(z,x)h(x)\chi\{x \in S/\Delta\} < \infty.$$

This tends to be the most difficult condition to check. In our case, this is equivalent to checking

$$\sum_{x>0} \pi_Q(x,0,1)\alpha^x < \infty \tag{5}$$

where π_Q is the invariant distribution for the Markov chain Q. Once again, the details of the proof will be omitted, but it turns out that using the work of [8], we can check (5) by finding a Lyapunov function V such that

$$KV - V \leq -f + s$$

where $f(x,y,z) = \alpha^x\chi\{x>0, y=0, z=1\}$, $s(x,y,z) = C'\chi\{x=y=0, z=2\}$, $C' > 0$ and $\chi(\cdot)$ is the indicator function. One can verify by direct substitution that one can choose constants $u, v > \alpha$ and $C, w > 0$ such that

$$V(x,y,z) = Cu^xv^yw^{z-1}$$

is a valid Lyapunov function.

Condition 7. This involves checking a uniform integrability condition, here we simply state that given $\hat{a}(\cdot) \equiv 1$, the condition is trivially satisfied.

Thus we can assert the results of Theorem 1. In its statement, there is a constant g in several of the expressions. We have now developed enough notation to give its definition.

$$g = \pi_W(0,0,2)\lambda_2\rho^{-1}H(1,0,2) + \sum_{x=1}^{\infty}\pi_W(x,x,1)\lambda_2\rho^{-(x+1)}H(x+1,x,1) \tag{6}$$

where $H(x,y,i)$ is the probability that starting at (x,y,i), $\mathcal{W}^\infty$ never hits Δ.

4.2 Non-single-class dominant (Theorem 2)

Now we expect the event F_ℓ to be reached through the effects of both classes of customer. Here we define $W[n] = (\tilde{W}[n], \hat{W}[n])$ to be the ordered pair $(Q_1[n] + Q_2[n], Q_1[n])$ and $\hat{W}[n]$ to simply be $Z[n]$. The transition kernel for this chain is the same as that for the single-class dominant case.

The main difference in the analysis comes from the change in structure of W^∞. We now effectively remove the zero boundary for both queues and always serve one customer at each queue and allow a negative number of customers at both queues. This translates to Δ being $(0,0,2)$, $(x,x,1)$ and $(y,0,2)$ where $x, y \in Z_+$.

The harmonic function remains unchanged, $h(x) = \rho^{-\tilde{x}_1}$ and we construct the twisted free chain in exactly the same manner as for the single-class dominant case. The fact that the harmonic function has remained unchanged means that in checking technical Conditions 0-7, the only conditions which are different are Conditions 1 and 6.

Condition 1. It is not difficult to see that $\varphi(1) = \varphi(2) = 1/2$. The server is simply serving one customer at each queue, disregarding zero boundaries.

Condition 6. Here we must check that (5) holds in addition to the condition

$$\sum_{y>0}\pi_Q(0,y,2)\alpha^y < \infty. \tag{7}$$

No additional analysis is required to show (7) as if (5) holds, then (7) holds by interchanging class 1 and class 2.

Finally, we give the constant g for this case:

$$\begin{aligned} g &= \pi_W(0,0,2)\lambda_2\rho^{-1}H(1,0,2) + \sum_{x=1}^{\infty}\pi_W(x,x,1)\lambda_2\rho^{-(x+1)}H(x+1,x,1) \\ &\quad + \sum_{y=1}^{\infty}\pi_W(y,0,2)\lambda_1\rho^{-(y+1)}H(y+1,1,2) \end{aligned} \tag{8}$$

where $H(x,y,i)$ is the probability that starting at (x,y,i), $\mathcal{W}^\infty$ never hits Δ.

5 Future Work

We are currently examining the following two generalizations: unequal service times as well as general, k_i-limited service policies (the number served on each visit to queue i may not exceed k_i.) This work is essentially complete. It is not conceptually more difficult than the work presented here, but is of considerably more technical difficulty.

In the future we hope to examine the impact of these results on buffer allocation problems, the effects of the arrival process (generalizing from simple Poisson processes to Markov modulated Poisson processes) and the impact of setup times on asymptotic behavior.

References

[1] O.J. Boxma and H. Takagi (Eds.). Special issue on polling systems. *Queueing Systems*, 11, 1992.

[2] W. Chang, D.G. Down, and R.D. Foley. Exact asymptotics for k_i-limited polling models. Working paper.

[3] G.L. Choudhury and W. Whitt. Computing distributions and moments in polling models by numerical transform inversion. *Performance Evaluation*, 25:267–292, 1996.

[4] N.G. Duffield. Exponents for the tails of distributions in some polling models. *Queueing Systems*, 26:105–119, 1997.

[5] R.D. Foley and D. McDonald. Join the shortest queue: stability and exact asymptotics. *Annals of Applied Probability.* Submitted. An electronic version is available at http://www.isye.gatech.edu/faculty/Robert_Foley/pub.html.

[6] H. Levy and M. Sidi. Polling systems: Applications, modeling, and optimization. *IEEE Trans. Comm.*, 38:1750–1760, 1990.

[7] D. McDonald. Asymptotics of first passage times for random walk in a quadrant. *Annals of Applied Probability*, 9:110–145, 1999.

[8] S. P. Meyn and R. L. Tweedie. *Markov Chains and Stochastic Stability.* Springer-Verlag, London, 1993.

Invariant rate functions for discrete time queues

A. J. Ganesh
Microsoft Research
St. George House, 1 Guildhall St.
Cambridge CB2 3NH, U.K.
ajg@microsoft.com

Neil O'Connell
BRIMS, Hewlett-Packard Labs
Filton Road, Stoke Gifford
Bristol BS12 6QZ, U.K.
noc@hplb.hpl.hp.com

Balaji Prabhakar
Dept. of Electrical Engineering, Stanford University
161 Packard, 350 Serra Mall, Stanford, CA 94305-9505
balaji@isl.stanford.edu

Abstract

We consider a discrete time queue with general service distribution and characterize a class of arrival processes whose large deviation rate function remains unchanged in passing through the queue. We show that this invariant rate function corresponds to an *exponential tilting* of the service distribution. We also establish a large deviations analogue of quasi-reversibility for such arrival processes.

1 Introduction

Burke's theorem says that if the arrival process to a $\cdot/M/1$ queue is Poisson with rate less than the service rate, then the departure process in equilibrium is also Poisson with the same rate. In other words, a Poisson process of rate α is a fixed point of the $\cdot/M/1$ queue with service rate 1, for every $\alpha < 1$. It has recently been shown [10] that a similar result holds for single-server queues with a general service time distribution: non-trivial stationary ergodic fixed points do exist. However, little is known about the properties of fixed points.

In this paper we consider the discrete time analogue of this problem and show under reasonable conditions that, if the arrival process satisfies a sample path large deviation principle (LDP) with a rate function given by a kind of exponential tilting of the service distribution, then the departure process satisfies a sample path LDP with the same rate function. In some sense, the fixed point is as similar in relative entropy to the service process as it can be, subject to its rate constraint, although any attempt to make sense of this interpretation raises more questions and seems to be an interesting topic for future research. We also find that an analogue of the quasi-reversibility property holds for arrival processes with invariant rate functions: the joint rate function for the current queue length and the past of the departure process, for the system in equilibrium, is equal to the sum of the indvidual rate functions. For completness, we also establish that the results of Mairesse and Prabhakar [10] on the existence of fixed points for continuous time queues carry over to the discrete setting.

The results in this paper are derived in the context of a discrete time queueing model which we now describe. The queue has arrival process $\{A_n, n \in \mathbb{Z}\}$, where A_n denotes the amount of work arriving in the n^{th} time slot. The service process is denoted by $\{S_n\}$, where S_n denotes the maximum amount of work that can be completed in the n^{th} time slot. The arrival and service processes are assumed to be stationary and ergodic sequences of positive real random variables. The workload process, $\{W_n\}$, is described by Lindley's recursion: $W_{n+1} = \max\{W_n + A_n - S_n, 0\}$. The amount of work departing in time slot n is given by

$$D_n = A_n + W_n - W_{n+1} = \min\{W_n + A_n, S_n\}. \tag{1}$$

If A_n and S_n are integer-valued for all n, then W_n can be thought of as the number of customers in the queue at time n.

In the next section we will present the relevant large deviation results from [11] and identify the form that the rate function of the arrivals process must take in order to be preserved by the queueing operator. In section 3 we present some results on the existence of fixed points. We show that the continuous-time results of [10] can be reproduced in discrete time with minor modifications.

2 Invariant rate functions for the single server queue

Let $\mathcal{X}$ be a Hausdorff topological space with Borel σ-algebra $\mathcal{B}$, and let X_n be a sequence of random variables taking values in $\mathcal{X}$. A *rate function* is a non-negative lower semicontinuous function on $\mathcal{X}$. We say that the sequence μ_n satisfies the *large deviation principle* (LDP) with rate function I, if for all $B \in \mathcal{B}$,

$$-\inf_{x \in B^\circ} I(x) \leq \liminf_n \frac{1}{n} \log P(X_n \in B) \leq \limsup_n \frac{1}{n} \log P(X_n \in B) \leq -\inf_{x \in \bar{B}} I(x).$$

Here B° and $\bar{B}$ denote the interior and closure of B, respectively. A large deviation rate function is *good* if it has compact level sets.

Let $\tilde{S}_n$ denote the polygonal approximation to the scaled service process, defined for $t \geq 0$ by:

$$\tilde{S}_n(t) = \hat{S}_n(t) + \left(t - \frac{\lfloor nt \rfloor}{n}\right)\left(\hat{S}_n\left(\frac{\lfloor nt \rfloor + 1}{n}\right) - \hat{S}_n\left(\frac{\lfloor nt \rfloor}{n}\right)\right), \quad \hat{S}_n(t) = \frac{1}{n}\sum_{k=1}^{\lfloor nt \rfloor} S_k.$$

Given an arrival process A_n, we define $\tilde{A}_n(t)$ analogously. Let $\mathcal{C}(\mathbb{R}_+)$ denote the space of continuous functions on the positive real line and $\mathcal{AC}(\mathbb{R}_+)$ the subset of absolutely continuous functions. We now record some hypotheses.

Assumptions:

1. The sequences $\{A_n\}$ and $\{S_n\}$ are stationary and ergodic, and independent of each other. The limiting cumulant generating functions,

$$\Lambda_A(\theta) = \lim_{n \to \infty} \frac{1}{n} \log E \exp \theta(A_1 + \ldots + A_n),$$

$$\Lambda_S(\theta) = \lim_{n \to \infty} \frac{1}{n} \log E \exp \theta(S_1 + \ldots + S_n),$$

exist as extended real numbers for all $\theta \in \mathbb{R}$ and are differentiable at the origin.

2. The sequences $\{\tilde{A}_n\}$ and $\{\tilde{S}_n\}$ satisfy the LDP in $\mathcal{C}(\mathbb{R}_+)$ equipped with the topology of uniform convergence on compacts, with respective rate functions $\mathcal{I}_A$ and $\mathcal{I}_S$ given by:
$$\mathcal{I}_A(\phi) = \begin{cases} \int_0^\infty I_A(\dot{\phi}(t))dt, & \text{if } \phi \in \mathcal{AC}(\mathbb{R}_+), \\ +\infty, & \text{otherwise,} \end{cases}$$
where
$$I_A(x) = \sup_{\theta \in \mathbb{R}} \{\theta x - \Lambda_A(\theta)\}$$
is the convex dual of Λ_A; $\mathcal{I}_S$ and I_S are described similarly in terms of Λ_S.
3. The stability condition $\Lambda'_A(0) < \Lambda'_S(0)$ holds.
4. $I_A(x) \leq I_S(x)$ for all $x \leq \Lambda'_A(0)$.

It has been shown by a number of authors under different levels of generality (see, for example, [1, 3, 4, 7]) that the tail of the workload distribution in equilibrium satisfies

$$\lim_{b \to \infty} \frac{1}{b} \log P(W > b) = -\delta, \tag{2}$$

where the decay rate δ is given by

$$\delta = \sup\{\theta : \Lambda_A(\theta) + \Lambda_S(-\theta) \leq 0\}. \tag{3}$$

Let $\tilde{D}_n(t)$ denote the scaled departure process, defined analogous to $\tilde{A}_n(t)$ and $\tilde{S}_n(t)$, and let $\tilde{W}_n(t) = W(\lfloor nt \rfloor)/n$ denote the scaled workload at time $\lfloor nt \rfloor$. A modification of [11, Theorem 3.3] yields a joint LDP for $\tilde{D}_n(1)$, the mean departure rate over the period $[0, n]$ and $\tilde{W}_n(1)$, the scaled workload at time n. We have

Theorem 1 *Under assumptions 1-3,* $\{(\tilde{D}_n(1), \tilde{W}_n(1)), n \in \mathbb{N}\}$ *jointly satisfy an LDP in* $\mathbb{R}^2$ *with rate function* $I_{D,W}$ *given by*

$$I_{D,W}(z, w) = \min\{\inf_{q \in C_1} f_1(q), \inf_{x \in C_2} f_2(x)\}, \tag{4}$$

where $x = (q, z_1, z_2, \beta)$,

$$\begin{aligned} f_1(q) &= \delta q + I_A(z + w - q) + I_S(z), \quad C_1 = \{q : 0 \leq q \leq z + w\}, \\ f_2(x) &= \delta q + \beta\Big[I_A\big(\frac{z_1 - q}{\beta}\big) + I_S\big(\frac{z_2}{\beta}\big)\Big] + \\ &\qquad (1-\beta)\Big[I_A\big(\frac{z + w - z_1}{1 - \beta}\big) + I_S\big(\frac{z - z_1}{1 - \beta}\big)\Big], \\ C_2 &= \{x : 0 \leq q \leq z_1 \leq z, z_2 \geq z_1, \beta \in [0, 1]\}. \end{aligned} \tag{5}$$

We omit a proof of this result but provide a brief sketch of the intuition behind it, which involves identifying the most likely path by which we can have departures at mean rate z on $[0, n]$ and workload nw at time n. Any sample path resulting in this event can be divided into three phases: from time 0 until the queue first becomes empty, from the last time the queue is empty on $[0, n]$ until time n and the period between when the queue is first empty and when it is last empty on $[0, n]$. One or more of these phases may be empty. It can be shown by the convexity of I_A and I_S that, on the optimal path, the arrival and service rates must be constant during each of these three phases. A little more analysis shows that the optimal path can only be of one of the following two types. In the first case, we have an initial workload nq at time 0, arrivals at constant rate $z+w-q$ and constant service capacity z over the entire period $[0,n]$. The queue never empties on $[0,n]$ and no service capacity is wasted. In the second scenario, the optimal path has two distinct phases. The first phase begins at time 0 with workload nq. The arrival rate is $(z_1-q)/\beta$ and the service capacity is z_2/β during this phase, which runs until time βn. Moreover, $z_1 \leq z_2$, and so the queue is empty at the end of the first phase. During the second phase, which runs over $[\beta n, n]$, the arrival rate is $(z+w-z_1)/(1-\beta)$, the service rate is $(z-z_1)/(1-\beta)$ and the queue is never empty. The optimization problem in the statement of the theorem corresponds to determining the most likely path within these scenarios.

It was shown in [5] that, if assumption 4 is violated, then the rate function governing the sample path LDP for the scaled departure process in equilibrium, $\tilde{D}_n$ (defined analogous to $\tilde{A}_n, \tilde{S}_n$), is not convex; in particular, there is no convex function $I(\cdot)$ such that $\mathcal{I}_D(\phi) = \int I(\dot{\phi})(t)dt$ for $\phi \in \mathcal{AC}(\mathbb{R}_+)$. Assumption 4 guarantees that I_D is convex and that, conditional on $\tilde{D}_n(1) = d$, $P(\sup_{t\in[0,1]} |\tilde{D}_n(t) - dt| > \epsilon) \to 0$ as $n \to \infty$ for every $\epsilon > 0$. This 'linear geodesic property' is still not sufficient to guarantee that the rate function $\mathcal{I}_D(\phi) = \int I(\dot{\phi})(t)dt$ for all $\phi \in \mathcal{AC}(\mathbb{R}_+)$. The main result of this section is that, given a service process which satisfies assumptions 1 and 2, we can find an arrival process such that assumptions 1-4 are satisfied, and such that the departure process satisfies the sample path LDP with rate function $\mathcal{I}_D = \mathcal{I}_A$. For this arrival process, we also show that a large deviations version of quasi-reversibility holds: the joint rate function for $\tilde{D}_n([0,1]), \tilde{W}_n(1)$ is the sum of the individual rate functions for $\tilde{D}_n([0,1])$ and $\tilde{W}_n(1)$ respectively. We state the result following some definitions.

Let $f : \mathbb{R} \to \mathbb{R} \cup \infty$ be a convex function. The effective domain of f, which we denote by dom f, is the set $\{x \in \mathbb{R} : f(x) < \infty\}$. For $x \in$ dom f, the subdifferential of f at x, denoted subdiff $f(x)$, is the set

$$\{\beta \in \mathbb{R} : \ f(y) \geq f(x) + \beta(y-x) \ \ \forall \, y \in \mathbb{R}\} .$$

Theorem 2 *Suppose the service process $\{S_n, n \in \mathbb{Z}\}$ satisfies assumptions 1 and 2, and assume without loss of generality that the mean service rate $E[S_1] = \Lambda_S'(0) = 1$. Let $\alpha \in (0,1)$ be in the interior of the effective domain of I_S. Define*

$$\lambda = \inf\{\mathit{subdiff}\ I_S(\alpha)\}. \tag{6}$$

If the arrival process $\{A_n, n \in \mathbb{Z}\}$ satisfies assumptions 1 and 2 and

$$I_A(x) = I_S(x) - I_S(\alpha) - \lambda(x-\alpha), \tag{7}$$

then assumptions 3 and 4 hold as well, and for any $t > 0$, $(\tilde{D}_n([0,t]), \tilde{W}_n(t))$ jointly satisfy the LDP in $C([0,t]) \times \mathbb{R}$ with good convex rate function

$$\mathcal{I}_{D,W}(\phi, w) = \int_0^t I_A(\dot{\phi}(s))ds \ + \ \delta w.$$

Note that λ exists and is finite by the convexity of I_S and the assumption that α is in the interior of dom I_S. It can also be verified that I_A defined by (7) is a rate function and that it is convex. Finally, if the service process is iid, as we assume in the next section, then there is an arrival process which satisfies the requirements of the theorem, namely the iid process corresponding to an exponential tilting of the service process. In other words, if μ_A and μ_S denote the probability distribution of A_1 and S_1 respectively, then $\mu_A \ll \mu_S$ and

$$\frac{d\mu_A}{d\mu_S}(x) = \frac{e^{\lambda x}}{\int_{\mathbb{R}} e^{\lambda z} d\mu_S(z)}, \quad x \in \mathbb{R}.$$

A continuous time queue is called quasi-reversible (see, e.g., [8]) if, in stationarity, the state of the queue at any time t, the departure process before time t and the arrival process after time t are mutually independent (the state is the same as the queue length if service times are exponential but is more complex in general).

In Theorem 2, we show that an analogue of this property, which we shall refer to as LD quasi-reversibility, holds for a general discrete time queue whose input has the invariant rate function given by (7). Specifically, the past of the departure process is independent of the current workload on a large deviations scale, in the sense that the joint rate function for the past departures and the current workload is the sum of their individual rate functions. We have from the definition of $\mathcal{I}_A, \mathcal{I}_S$ that the joint rate function for $(\tilde{A}_n((-\infty, t]), \tilde{S}_n((-\infty, t]), \tilde{A}_n(t, \infty)$ decomposes into a sum of their individual rate functions. Since the workload at t and the departures up to time t depend only on the arrivals and services up to time t, we see that in fact the past departures, the current workload, and the future arrivals are mutually independent on the large deviation scale, in the sense described above.

The proof of Theorem 2 relies on first establishing the LDP jointly for the mean departure rate, $\tilde{D}_n(t)$, over some period $[0, nt]$ and the workload, $\tilde{W}_n(t)$, at time nt. This LDP is then extended to a joint LDP for the vector $(\tilde{D}_n(t_1), \ldots, \tilde{D}_n(t_k), \tilde{W}_n(t_k)$, for arbitrary $0 < t_1 < \ldots < t_k = 1$. The property of LD quasi-reversibility plays a crucial role in this extension. Finally, this collection of LDPs is lifted to a sample path LDP for $\tilde{D}_n([0,1])$ using the method of projective limits; this step closely parallels the proof of Mogulskii's theorem in [2]. The key steps are encapsulated in the lemmas which we state below, without proof. The details can be found in [6].

Lemma 1 *Let* $\{A_n\}$, $\{S_n\}$ *satisfy the assumptions of Theorem 2, with* I_A *given by (6) and (7). Then, for* δ *defined by (3),* $\delta = -\lambda$.

An implication of this result is that, if the arrival process is as in the statement of Theorem 2, then along the most likely path leading to the build-up of large workloads, the empirical distribution of the arrivals is close to the service distribution and *vice versa*. This interchange of roles between arrival and service processes is known to characterize the build-up of large workloads in the $M/M/1$ queue.

Lemma 2 *Suppose* $\{A_n\}$, $\{S_n\}$ *satisfy the assumptions of Theorem 2, with* I_A *given by (6) and (7). Let* $z, w \geq 0$ *be given. Then,*

$$f_1(q) \geq I_A(z) + \delta w \quad \text{and} \quad f_2(x) \geq I_A(z) + \delta w$$

for any $q \in C_1$ *and any* $x \in C_2$.

Lemma 3 *Let $w, z \geq 0$ be given. If $z + w \geq 1$, then the infimum in (4) is achieved by f_1 at $q^* = z + w - 1$, whereas, if $z + w \leq 1$, then the infimum in (4) is achieved by f_2 at*

$$x^* = (q, z_1, z_2, \beta) = \Big(0, \frac{z(1-w-z)}{1-z}, \frac{1-w-z}{1-z}, \frac{1-w-z}{1-z}\Big).$$

In either case, the minimum value, $I_{D,W}(z, w)$, is $I_A(z) + \delta w$.

Lemma 3 establishes an LD quasi-reversibility property: the joint rate function for the mean departure rate on $(0, n)$ and the workload at time n is the sum of the corresponding individual rate functions. In other words, the queue is approximately in equilibrium at time n (the rate function for the workload is the same as the equilibrium rate function) irrespective of the mean rate of departures on $(0, n)$. This property turns out to be crucial to the proof of Lemma 4 below and thereby to the proof of Theorem 2.

Lemma 4 *For any $k \in \mathbb{N}$ and $0 < t_1 < \ldots < t_k$, $(\tilde{D}_n(t_1), \ldots, \tilde{D}_n(t_k), \tilde{W}_n(t_k))$ jointly satisfy the LDP in $\mathbb{R}^{k+1}$ with rate function*

$$I^k_{D,W}(z_1, \ldots, z_k, w) = \sum_{i=1}^{k} (t_i - t_{i-1}) I_A\Big(\frac{z_i - z_{i-1}}{t_i - t_{i-1}}\Big) + \delta w.$$

3 Existence of fixed points

In this section we present some results on the existence of fixed points in a discrete-time setting, using arguments analagous to those presented in [10] for the continuous time setting.

Consider the space $\mathbb{R}^{\mathbb{Z}}$ equipped with the topology of coordinatewise convergence, which is metrizable using the metric

$$d(\mathbf{x}, \mathbf{y}) = \sum_{i \in \mathbb{Z}} \frac{1}{2^{|i|}} \frac{|x_i - y_i|}{1 + |x_i - y_i|}.$$

We let M be the space of stationary probability measures on $\mathbb{R}^{\mathbb{Z}}$ which are stochastically dominated by the service process and equip it with the weak topology generated by the metric $d(\cdot, \cdot)$. Clearly, M is convex, and it can be shown to be a compact metric space. We denote by M_e the subset of M consisting of ergodic measures and by M^α (resp. M^α_e) the subset consisting of measures (resp. ergodic measures) whose one-dimensional marginals have mean $\alpha \in \mathbb{R}$.

Consider an infinite queueing tandem. Let A_n denote the amount of work entering the first queue of the tandem in time slot n and let S^k_n denote the amount of work that can be served by queue k in time slot n, $k \in \mathbb{N}$, $n \in \mathbb{Z}$. Let W^k_n denote the workload in queue k at the beginning of time slot n and D^k_n the amount of work departing queue k and entering queue $(k+1)$ during time slot n. We assume the following in the remainder of this section.

Assumptions S^k_n is an iid sequence for each fixed k and identically distributed for all k,

$$ES^1_1 = 1, \quad \Lambda_S(\theta) := \log\, E \exp \theta S^1_1 < \infty \quad \text{for all } \theta \text{ in a neighbourhood of } 0.$$

The service distribution is non-degenerate, i.e., $P(S_1^1 \neq 1) > 0$. The arrival process A_n and the service processes S_n^k at the different queues are mutually independent, A_n is stationary and ergodic with rate $\alpha < 1$, i.e.,

$$\lim_{n\to\infty} \frac{1}{n}\sum_{i=1}^{n} A_i = E[A_1] = \alpha \quad a.s.,$$

and A_n is stochastically dominated by the service process (at any queue). In addition,

$$\Lambda_A(\theta) := \lim_{n\to\infty} \frac{1}{n} \log E \exp\ \theta(A_1 + \ldots + A_n)$$

exists as an extended real number for all $\theta \in \mathbb{R}$, Λ_A is differentiable in the interior of its domain (the set where Λ_A is finite) and steep, i.e., $|\Lambda_A'(\theta)| \to \infty$ as θ approaches the boundary of its domain.

It follows from the above assumptions that the departure process $(D_n^k, n \in \mathbb{Z})$ is stationary and ergodic with rate α, for each $k \in \mathbb{N}$.

Let $M_0 \subset M$ be the set of stationary probability measures on $\mathbb{R}^{\mathbb{Z}}$ whose ergodic decompositions do not contain an atom at the service distribution. We can define the queueing operator $\mathcal{Q}$ on M_0 by setting $\mathcal{Q}(\nu)$ to be the law of the departure process corresponding to an arrival process which is independent of the service process and has law $\nu \in M_0$. It follows from Loynes' construction [9] that $\mathcal{Q}$ is well-defined and maps M_0 into itself, that it preserves ergodicity and that it is mean-preserving. Moreover, $\mathcal{Q}$ is linear, i.e.,

$$\mathcal{Q}(\beta\nu_1 + (1-\beta)\nu_2) = \beta\mathcal{Q}(\nu_1) + (1-\beta)\mathcal{Q}(\nu_2),$$

for all $\nu_1, \nu_2 \in M_0$ and $\beta \in [0,1]$. Finally, $\mathcal{Q}$ is continuous in the weak topology restricted to M_0. The proof of the last statement is virtually identical to that of [10, Theorem 4.3].

Let μ_0 denote the law of $(A_n, n \in \mathbb{Z})$, μ_k the law of $(D_n^k, n \in \mathbb{Z})$ and μ^S the law of the service process, $(S_n, n \in \mathbb{Z})$. We have assumed that $\mu_0 \in M_e^\alpha$ for some $\alpha < 1$, whereas $\mu^S \in M_e^1$, so μ_0 is not the service distribution. Since μ_0 consists of a single ergodic component, it follows that $\mu_0 \in M_0$. Hence, so is $\mu_k = \mathcal{Q}^k(\mu_0)$ for any $k \in \mathbb{N}$, where $\mathcal{Q}^k$ denotes the k^{th} iterate of $\mathcal{Q}$. Since M_0 is clearly convex,

$$\lambda_k := \frac{1}{k}\sum_{i=0}^{k-1} \mu_k \in M_0 \quad \text{for all } k \in \mathbb{N}.$$

Since M is compact, there is a subsequence $k(j)$ of $\mathbb{N}$ such that $\lambda_{k(j)} \to \lambda$ for some $\lambda \in M$. We have

Theorem 3 *Let $\lambda \in M$ be defined as above as a subsequential limit of the λ_k's, where λ_k is the Cesaro average of the distributions of the departures from the first k queues in the tandem. Then $\lambda \in M_0$ and $\mathcal{Q}(\lambda) = \lambda$, i.e., λ is a fixed point of the queueing operator.*

Proof: Since $\lambda_k \in M_0$ and $\mathcal{Q} : M_0 \to M_0$, we have $\mathcal{Q}(\lambda_k) \in M_0$ for all k. But,

$$\mathcal{Q}(\lambda_k) = \mathcal{Q}\left(\frac{1}{k}\sum_{i=0}^{k-1} \mu_i\right) = \frac{1}{k}\sum_{i=1}^{k} \mu_i = \lambda_k + \frac{1}{k}(\mu_k - \mu_0). \tag{8}$$

To obtain the second equality, we have used the fact that $\mathcal{Q}$ is linear and that $\mathcal{Q}(\mu_i) = \mu_{i+1}$ by definition of the μ_i. It is clear from (8) that,

$$\lim_{j\to\infty} \mathcal{Q}(\lambda_{k(j)}) = \lim_{j\to\infty} \lambda_{k(j)} = \lambda. \tag{9}$$

We have from Lemma 7 below that $\lambda \in M_0$. Since $\mathcal{Q} : M_0 \to M_0$ is continuous in the weak topology and $\lambda_{k(j)} \to \lambda$ in this topology, it follows that

$$\lim_{j\to\infty} \mathcal{Q}(\lambda_{k(j)}) = \mathcal{Q}(\lambda). \tag{10}$$

By (9) and (10), $\mathcal{Q}(\lambda) = \lambda$. ■

Lemma 5 *Consider a sequence of stationary arrival distributions $\nu_k \in M$, converging weakly to a stationary arrival distribution $\nu \in M$. Let $W_0(k)$ (resp. W_0) denote a random variable with the distribution of the workload at the beginning of time slot zero, when the arrival process has distribution ν_k (resp. ν) and is independent of the service process. Then, we have*

$$\liminf_{k\to\infty} E[W_0(k)] \geq E[W_0].$$

The result holds even if $E[W_0] = +\infty$.

The proof is along the lines of the proof of [10, Lemma 4.4] and is omitted.

Lemma 6 *Let $W_0(k)$ denote a random variable with the distribution of the workload at the beginning of time slot zero, when the arrival process has distribution λ_k and the service process has distribution μ^S. Then, we have*

$$\limsup_{k\to\infty} E[W_0(k)] < +\infty.$$

The proof is omitted for brevity; details can be found in [6].

Lemma 7 *The distribution λ, which was defined in the statement of the theorem as a subsequential limit of the λ_k's (mixtures of departure distributions from successive queues in the tandem), does not contain an atom at the service distribution. In other words, $\lambda \in M_0$.*

Proof: Since the service process was assumed to be non-deterministic, it follows from Loynes' construction that if the arrival process is independent of the service process but has the same distribution, then the expected workload at time zero is infinite. By the linearity of the queueing operator, the same is true if the ergodic decomposition of the arrival distribution contains an atom at the service distribution. In other words, if W_0 denotes the workload at time zero when the arrival process has distribution λ and is independent of the service process, then

$$\lambda \in M \setminus M_0 \quad \Rightarrow \quad E[W_0] = +\infty.$$

Now $\lambda_{k(j)} \to \lambda \in M$ by definition, so it follows from Lemmas 5 and 6 that $E[W_0] < +\infty$. Hence $\lambda \in M_0$. ■

4 Conclusions

We consider a discrete time, single server queue with general service distribution and establish the existence of non-trivial stationary fixed points. We also show that there is a class of arrival processes, independent of the service process, such that the arrival and departure processes satisfy a sample path LDP with the same rate function. For these arrival processes, a large deviations analogue of quasi-reversibility holds. The invariant rate functions correspond to a kind of exponential tilting of the service process, so that they are, in some sense, as close in relative entropy to the service distribution as possible, subject to a rate constraint.

References

[1] C.S. Chang, "Stability, queue length and delay of deterministic and stochastic queueing networks," *IEEE Trans. Autom. Control* vol. 39, pp. 913-931, 1994.

[2] A. Dembo and O. Zeitouni, *Large Deviations Techniques and Applications*, Jones and Bartlett, 1993.

[3] G. de Veciana and J. Walrand, "Effective bandwidths: call admission, traffic policing and filtering for ATM networks," *Queueing Systems*, vol.20, pp. 37-59, 1995.

[4] N. Duffield and Neil O'Connell, "Large deviations and overflow probabilities for the general single server queue, with applications," *Math. Proc. Cambridge Phil. Soc.* vol. 118, no. 1, 1995.

[5] A. J. Ganesh and Neil O'Connell, "The linear geodesic property is not generally preserved by a FIFO queue," *Ann. Appl. Probab.*, vol. 8, pp. 98-111, 1998.

[6] A. J. Ganesh, Neil O'Connell and Balaji Prabhakar, "Fixed points for discrete time queues and their large deviations behaviour," in preparation.

[7] Peter W. Glynn and Ward Whitt, "Logarithmic asymptotics for steady-state tail probabilities in a single-server queue," *J. Appl. Prob.*, vol. 31A, pp. 131-156, 1994.

[8] F.P. Kelly, *Reversibility and Stochastic Networks*, John Wiley & Sons, New York, 1979.

[9] R. Loynes, "The stability of a queue with non-independent interarrival and service times," *proc. Camb. Phil. Soc.* vol. 58, pp. 497-520, 1962.

[10] Jean Mairesse and Balaji Prabhakar, "On the existence of fixed points for the $\cdot/GI/1/\infty$ queue," preprint, 1999.

[11] Neil O'Connell, "Large deviations for departures from a shared buffer," *J. Appl. Prob.*, vol. 34, no. 3, pp. 753-766, 1997.

Large Deviations and Optimality of the Largest Weighted Delay First Discipline

Alexander L. Stolyar and Kavita Ramanan
Bell Labs, Lucent Technologies
600 Mountain Avenue
Murray Hill, NJ 07974
{stolyar,kavita}@research.bell-labs.com

Abstract

We consider a single server system with N input flows. We assume that each flow has stationary increments and satisfies a sample path large deviation principle. We introduce the Largest Weighted Delay First (LWDF) queueing discipline associated with any given weight vector $\alpha = (\alpha_1, \ldots, \alpha_N)$. We show that under the LWDF discipline the sequence of scaled stationary distributions of the delay $\hat{w}_i$ of each flow satisfy a large deviation principle with the rate function given by a finite-dimensional optimization problem. We also prove that the LWDF discipline is optimal in the sense that it maximizes the quantity

$$\min_{i=1,\ldots,N} \left[\alpha_i \lim_{n\to\infty} \frac{-1}{n} \log P\left(\hat{w}_i > n\right)\right],$$

amongst a large class of work conserving disciplines.

1 Introduction

A very important and challenging problem in the design of high-speed communication networks is that of providing Quality of Service (QoS) guarantees, usually specified in terms of loss probabilities or delays of packets in the network. The control of delays is often of crucial importance, especially for real-time applications like video. For example, the QoS requirements for each traffic flow (customer class) can be specified in terms of a deadline T_i and an allowed violation probability δ_i. More precisely, if there are N classes of users and $\hat{w}_i$ is the stationary customer delay of the ith class, then one would like to determine if there exists a policy that would meet the given QoS constraints

$$P(\hat{w}_i > T_i) \le \delta_i \quad \text{for } i = 1, \ldots, N. \tag{1.1}$$

In this paper we restrict ourselves to a single node and address a related asymptotic question. Suppose a set of positive constants (weights) $\alpha_1, \alpha_2, \ldots, \alpha_N$ is fixed. Withiout loss of generality let $\alpha_1 \le \alpha_2 \le \ldots \le \alpha_N$. For a non-negative random variable X, let us use the notation

$$\beta(X) \doteq \lim_{n\to\infty} -\frac{1}{n} \log P(X > n) ,$$

assuming the limit exists. We ask the following question. Is there a discipline G such that the *asymptotic* QoS requirements

$$\beta(\hat{w}_i^G) \geq \alpha_i^{-1} \quad \text{for } i = 1, \ldots, N \tag{1.2}$$

are satisfied, where $\hat{w}_i^G$ is the stationary delay for class i with discipline G? (Observe that when T_i is large and δ_i is small, the problem (1.1), can be related to (1.2) by setting $\alpha_i = -T_i / \log \delta_i$ for all i.) We address the problem (1.2) by considering the following equivalent *optimization problem*:

$$\max_G \min_{i=1,\ldots,N} \alpha_i \beta(\hat{w}_i^G) \, . \tag{1.3}$$

We now introduce the Largest Weighted Delay First (LWDF) discipline, defined as follows.

Definition 1.1 [LWDF] *The LWDF discipline associated with a set of weights $\alpha_1, \ldots, \alpha_N$, is a non-preemptive discipline that always chooses for service the head-of-the-line (i.e., longest waiting) customer of the class i for which the quantity $\hat{w}_i / \alpha_i$ is maximum, where $\hat{w}_i$ is the current delay of the head-of-the-line class i customer. By convention, in case of a tie, the class with the highest index is chosen.*

Under the assumption that the input flows have stationary increments and satisfy a sample path large deviation principle, and the system is stable, we prove that *LWDF is an optimal solution of the problem (1.3).* This means that if there is any discipline G which solves problem (1.2), then LWDF does so.

Note that if all weights α_i are equal, LWDF reduces to the simple FIFO discipline. Also note that *LWDF is invariant with respect to the stochastic structure of the input flows*, in the sense that the algorithm itself relies *only* on the weights α_i which characterize the QoS requirements of the flows, and not on the structure of the input flows.

To provide some intuition behind our result, note that if for a given discipline G all $\beta(\hat{w}_i^G)$ are well defined, then

$$\min_{i=1,\ldots,N} \alpha_i \beta(\hat{w}_i^G) = \beta(\max_{i=1,\ldots,N} \frac{\hat{w}_i^G}{\alpha_i}) \tag{1.4}$$

Thus problem (1.3) is reduced to maximizing (over all disciplines G) the rate of of decay of the tail of the stationary distribution of the *maximal weighted delay* $\hat{r}^G \doteq \max_{i=1,\ldots,N} \hat{w}_i^G / \alpha_i$. We solve this problem using large deviations techniques. First, we establish a large deviations principle (LDP) for the sequence of scaled stationary maximal weighted delays under the LWDF discipline. In particular, we characterize J_*, the quantity in (1.4) for the LWDF, in terms of finite-dimensional optimization problem. (In a similar fashion, the LDP for each sequence of scaled stationary delays $\hat{w}_i$ (under the LWDF discipline) can be obtained.) Then we establish optimality of the LWDF discipline by showing that the value of (1.4) for any other discipline G is bounded above by J_*.

Due to space limitations we do not present any proofs in this paper. In particular, we do not discuss crucial parts of our analysis related to the characterization of the *fluid process* under the LWDF discipline, and the identification of *optimal fluid paths* that characterize the large deviations of $\hat{r}^G$. We merely outline the technique used in the proof, and state some intermediate results. The complete development and all proofs can be found in [8].

There is a vast body of literature on QoS guarantees in communication networks (see [7] and references therein). In much of that work, a policy is fixed a priori and the focus is then on performance analysis. Our perspective is different: we fix the desired performance objective, and then determine the optimal policy to achieve it. Other work that takes this perspective generally focuses on optimality in the heavy traffic regime. In contrast, this paper provides a proof of optimality with respect to a large deviations criterion in a queueing context. Analysis of queueing processes, and in particular analysis of their large deviation properties, is usually further complicated by the discontinuities introduced by the state space constraints. Although the existence of large deviation principles for sequences of scaled queue length processes have been established for a relatively large class of queueing networks [2], the problem of determining the rate function and characterizing optimal paths has remained elusive for the vast majority of cases where there are multiple classes, even for a single node. Most of the results in the single node case have been limited to two classes [5, 6], with the multi-dimensional results being rather few and restricted to queue lengths [3], as opposed to delays.

The outline of the paper is as follows. The basic notation used throughout the paper is provided in Section 2. In Section 3 we state our basic assumptions and formulate the main result in Theorem 3.2. In Section 4 we rigorously define the stationary maximal weighted delay (which provides the precise meaning of our main result), and finally in Section 5 we outline the technique used to prove the main result.

2 Basic Notation and Definitions

Let $\mathcal{D}$ [respectively $\mathcal{D}_+$] be the space of RCLL functions (i.e. right continuous functions with left limits) on $(-\infty, \infty)$ [respectively $[0, \infty)$]. Unless otherwise specified, we assume $\mathcal{D}$ and $\mathcal{D}_+$ are endowed with the topology of uniform convergence on compact sets (u.o.c.). For any $s \geq 0$ and $f \in \mathcal{D}^N$, we define the norm

$$||f||_s \doteq \max_{i=1,\ldots,N} \sup_{-s \leq t \leq s} |f_i(s)|.$$

For $f \in \mathcal{D}_+^N$ the norm $||f||_s$ is defined similarly with $-s$ replaced by 0 in the above display. Thus convergence in $\mathcal{D}$ or $\mathcal{D}_+$ is equivalent to convergence in the corresponding norm $||\cdot||_s$ for all $s > 0$. For $f \in \mathcal{D}$ we use $f(t-)$ to denote the left limit $\lim_{u \uparrow t} f(u)$ of the function f at the point t. As measurable spaces, we always assume that $\mathcal{D}$ and $\mathcal{D}_+$ are endowed with the σ-algebra generated by the cylinder sets. We now define some subspaces of $\mathcal{D}$ and $\mathcal{D}_+$ that will be used often in the sequel. Let $\mathcal{S}$ be the space of functions in $\mathcal{D}$ which are non-decreasing, piecewise constant, and have only a finite number of jumps on any finite time interval, and let $\mathcal{S}_+$ be the subset of *non-negative* functions in $\mathcal{D}_+$ which are non-decreasing, piecewise constant, and have only a finite number of jumps on any finite time interval. Let $\mathcal{S}_{+,0}$ be the subset of functions h in $\mathcal{S}_+$ such that $h(0) = 0$. We use $\mathcal{I}$ to denote the subset of non-decreasing functions in $\mathcal{D}$, and $\mathcal{I}_+$ the subset of *non-negative* non-decreasing functions in $\mathcal{D}_+$. We denote by $\mathcal{I}_{+,0}$ the subset of functions h in $\mathcal{I}_+$ such that $h(0) = 0$. Let $\mathcal{C}$ be the subset of continuous functions in $\mathcal{I}_{+,0}$, and let $\mathcal{C}_a$ be the subset of absolutely continuous functions in $\mathcal{C}$. We assume that the subspaces inherit the topology and σ-algebra of the original space.

Let $\Omega \doteq (\Omega, \mathcal{F}, P)$ be a probability space. We assume that Ω is large enough to support all the independent random processes that we use in the paper. We denote by $P_*(B)$ the inner measure (with respect to the probability P) of an arbitrary subset $B \subseteq \Omega$. If $B \in \mathcal{F}$, then $P_*(B) = P(B)$.

We define the scaling operator Γ^c, $c > 0$, for elements $Y \in \mathcal{D}^N$ (or $\mathcal{D}^N_+$) as follows:

$$(\Gamma^c Y)(t) \doteq \frac{1}{c} Y(ct) \ . \tag{2.5}$$

For a scalar b $\Gamma^c b \doteq b/c$, and for the pair (b, Y), $\Gamma^c(b, Y) \doteq (b/c, \Gamma^c Y)$.

3 The Model and the Main Result

We consider a multi-class single server queueing system with N input flows that satisfy the following assumptions.

Assumption 3.1 (Input flows)

1. *Each flow* $\mathbf{f_i}$ *is a random process on* Ω *with stationary increments taking values in* $\mathcal{S}$.
2. *The flows* $\mathbf{f_i}$, $i = 1, \ldots, N$, *are mutually independent.*
3. *For each* i, *and every* $T < \infty$, *the sequence of processes* $\{\mathbf{f}^n_i - \mathbf{f}^n_i(0), n = 1, 2, \ldots\}$ *restricted to* $[0, T]$ *satisfies a LDP [1, p.5] with rate function* J^i_T *given by*

$$J^i_T(h) \doteq \begin{cases} \int_0^T L_i(h(s))ds & \text{if } h \in \mathcal{C}_a, \\ \infty & \text{otherwise,} \end{cases} \tag{3.6}$$

   where $\mathbf{f}^n = \Gamma^n \mathbf{f}$ *is the scaled version of* $\mathbf{f}$ *as defined in (2.5),* L_i *is a convex function taking values in* $[0, \infty]$ *such that* $L_i(\lambda_i) = 0$ *for some* $\lambda_i \in (0, \infty)$, $L_i(x) > 0$ *for* $x \neq \lambda_i$, *and* $\lim_{x \to \infty} L_i(x)/x = \infty$.

4. *The server is not overloaded. In other words,* $\sum_1^N \lambda_i < 1$.

The process $\mathbf{f_i}(t)$ represents the cumulative amount of work of class i that has entered the system by time t. A jump in $\mathbf{f_i}(\cdot)$ at time t corresponds to a "customer" arrival, with the "service time" of that customer equal to the size of the jump $\mathbf{f_i}(t) - \mathbf{f_i}(t-)$.

Definition 3.1 *Consider the class* $\mathcal{G}$ *of queueing (or scheduling) disciplines* G *such that*
(a) G *is work conserving*
(b) scheduling decisions at any time t *are* independent *of both the future of the process and the history of the process* before the beginning of the current busy period.

Let G be an arbitrary discipline in $\mathcal{G}$. Given an input flow sample path $f \in \mathcal{S}^N$, let $\hat{\tau}^G_i(t)$ be the arrival time of the "longest waiting" class i customer present in the system at time t, including the customer(s) being served at time t. (It can be shown that this function is well defined for almost all sample paths of the input flow process $\mathbf{f} = (\mathbf{f_1}, \ldots, \mathbf{f_N})$.) By convention we assume that $\hat{\tau}^G_i$ is right-continuous, and set $\hat{\tau}^G_i(t) = t$ if there is no class i customer in the system at time t. We refer to $\hat{\tau}^G_i \doteq (\hat{\tau}^G_i(t), -\infty < t < \infty)$ as the class i *backlog* process. Suppose we are given the set of positive *weights* $0 < \alpha_1 \leq \alpha_2 \leq \ldots \leq \alpha_N$. Then we define the class i *delay* $\hat{w}^G_i$ and *weighted delay* $\hat{r}^G_i$

processes, and the *maximal weighted delay* $\hat{r}^G$ process in terms of the class i backlog process as follows. For $t \geq 0$ and $i = 1, \ldots, N$,

$$\hat{w}_i^G(t) \doteq t - \hat{\tau}_i^G(t), \quad \hat{r}^G(t) \doteq \max_i \hat{w}_i^G(t)/\alpha_i$$

We are interested in identifying a discipline that is optimal in the sense that it maximizes the exponential decay rate of the stationary distribution of the maximal weighted delay $\hat{r}^G(\cdot)$. Our main result is that the Largest Weighted Delay First (LWDF) discipline introduced in Section 1 is optimal in a large deviations sense as made precise below in Theorem 3.2.

Theorem 3.2 *There exists $J_* < \infty$ such that the following holds.*

1. *For the LWDF discipline defined in Section 1*

$$\limsup_{n\to\infty} \frac{1}{n} \log P\left(\frac{1}{n}\hat{\mathbf{r}}(0) > 1\right) \leq -J_*, \tag{3.7}$$

where $\hat{r}(0)$ is the stationary maximal weighted delay associated with LWDF.

2. *For any $G \in \mathcal{G}$,*

$$\liminf_{n\to\infty} \frac{1}{n} \log P_*\left(\frac{1}{n}\hat{\mathbf{r}}^G(0) > 1\right) \geq -J_*\,, \tag{3.8}$$

where $\hat{r}^G(0)$ is the stationary maximal weighted delay associated with the discipline G.

3. *Moreover, J_* solves the following finite dimensional optimization problem*

$$J_* = \min_{j;x_1,\ldots,x_j} \frac{1}{\gamma} \sum_{i=1}^{j} (1 - \alpha_i\gamma) L_i(x_i), \tag{3.9}$$

subject to

$$j \in \{1, \ldots, N\}, \quad x_i > 0, \quad \sum_{i=1}^{j} x_i > 1,$$

and

$$\frac{1}{\alpha_{j+1}} < \gamma = \frac{\sum_{i=1}^{j} x_i - 1}{\sum_{i=1}^{j} \alpha_i x_i} \leq \frac{1}{\alpha_j},$$

with $\alpha_{N+1} \doteq \infty$ by convention.

Remark 1. A rigorous definition of the stationary maximal weighted delay for an arbitrary discipline $G \in \mathcal{G}$ is the subject of the next section. As we will see, $\hat{\mathbf{r}}^G(\cdot)$ need not in general be a measurable function on the probability space. To allow for this generality, we use the notion of inner measure in the lower bound (statement 2 above).
Remark 2. A result analogous to statements 1) and 2) of Theorem 3.2 holds if in the formulation we replace the maximal stationary weighted delay $\hat{\mathbf{r}} = \max_i \hat{\mathbf{w}}_i^G/\alpha_i$ by $\max_i \mathbf{q_i}^G/\alpha_i$, where $\mathbf{q_i}^G$ is the stationary unfinished work of class i. In fact, the analysis leading to this result is very similar and much simpler (because the fluid process for the unfinished work is continuous).

4 The Stationary Maximal Weighted Delay Process

4.1 Operators Associated with a Queueing Discipline

Consider a fixed discipline $G \in \mathcal{G}$. Suppose the server is empty at time 0, which means that the left limit of the unfinished work in the system is equal to 0. Let a path $f \in \mathcal{S}_+^N$ describe the input process starting at time 0. (If $f_i(0) > 0$, it means that a type i customer arrived at time 0 requiring a service time $f_i(0)$.) By assumption (b) in the definition of the class $\mathcal{G}$, for each i, $\hat{\tau}_i^G(0) = 0$ and the "evolution" of $\hat{\tau}_i^G(\cdot)$ from time 0 onwards depends only on the path f, and not on the system behavior before $t = 0$. Given G, we define

$$\hat{A}^G : \mathcal{S}_+^N \mapsto \mathcal{I}_{+,0}^N$$

to be the operator that maps f into the corresponding $\hat{\tau}^G \doteq ((\hat{\tau}_i^G(t),\ t \geq 0),\ i = 1,\dots,N)$. It is clear that for any $f \in \mathcal{S}_+^N$, $\hat{\tau}^G = \hat{A}^G f$ is such that for $i = 1,\dots,N$,

$$\hat{\tau}_i^G(0) = 0 \quad \text{and} \quad \hat{\tau}_i^G(t) \leq t \text{ for } t \geq 0.$$

Once again by assumpton (b) in the definition of the class $\mathcal{G}$, the discipline G is completely characterised by its "behavior" within a busy period. However the points in time when new busy periods of the system start do *not* depend on the queueing discipline, since all disciplines in $\mathcal{G}$ are work conserving. Thus the operator $\hat{A}^G$ associated with a discipline G completely characterizes the evolution of all $\hat{\tau}_i^G(t)$ for all $t \in (-\infty, \infty)$. Moreover, if the discipline $G \in \mathcal{G}$ is *non-preemptive*, then the operator $\hat{A}^G$ completely characterizes the discipline itself. We also introduce a related operator

$$\hat{R}^G : \mathcal{S}_+^N \mapsto \mathcal{I}_{+,0}$$

which maps an input flow path $f \in \mathcal{S}_+^N$ into the corresponding maximum weighted delay $\hat{r}^G \doteq (\hat{r}^G(t),\ t \geq 0)$. Clearly, each operator $\hat{R}^G$ is uniquely determined by the corresponding operator $\hat{A}^G$.

Note that LWDF discipline is First-In-First-Out within each class. We drop the superscript G for the LWDF discipline and denote the operators $\hat{A}^G$ and $\hat{R}^G$ associated with the LWDF discipline simply by $\hat{A}$ and $\hat{R}$ respectively.

4.2 The Stationary State Process

In this section we introduce an operator H that associates to any input path $f \in \mathcal{I}^N$ the state of the queueing system. The state is defined in such a way that it depends only on the input path f and is independent of the particular work conserving queueing discipline $G \in \mathcal{G}$ used. Roughly speaking, the state of the system at time t is the history of the input flows from the beginning of the current busy period. Using measurability properties of the operator H and the fact that the stochastic input flows have stationary increments, we then use a Loynes type construction [4] to define the stationary state process $\boldsymbol{\theta} \doteq H\mathbf{f}$. Finally, in Lemma 5.3 we also formulate a kind of LDP for the sequence of scaled state distributions.

We introduce the space of system states $\Psi \doteq \{\psi = (b, g) : b \in \mathbb{R}_+, g \in \mathcal{S}_+^N\}$. We endow Ψ with the natural topology generated by u.o.c. convergence, and with the σ-algebra generated by cylinder subsets. Let Θ be the space of RCLL functions on $(-\infty, \infty)$ taking values in Ψ. We define the deterministic mapping $H : \mathcal{S}^N \to \Theta$ that takes input flow paths $f \in \mathcal{S}^N$ to the system state sample paths as follows. Given any $f \in \mathcal{S}^N$, for each $t \in (-\infty, \infty)$, $Hf(t) \doteq (b(t), g) \in \Psi$, where

$$b(t) \doteq t - \sup\left\{s \le t : \left(\sum_{i=1}^{N} f_i(s-) - s\right) = z(t)\right\}, \tag{4.10}$$

$$z(t) \doteq \inf_{s \le t}\left(\sum_{i=1}^{N} f_i(s) - s\right), \tag{4.11}$$

and

$$g_i(s) \doteq \begin{cases} f_i(s + t - b(t)) - f_i((t - b(t))-), & 0 \le s \le b(t), \\ f_i(t) - f_i((t - b(t))-), & s > b(t). \end{cases} \tag{4.12}$$

When f is the realization of an input flow to a single server queueing system (with service rate 1), $b(t)$ represents the time elapsed from the start of the busy period in progress at time t, and g captures the history of the input flows from the start of the current busy period. Note that b and g have this intuitive interpretation only when $z(t)$ is finite for all t. However, since this is true for almost all input flows $\mathbf{f}$ satisfying Assumption 3.1, this does not pose a problem in the definition of the process $H\mathbf{f}$ given below.

Lemma 4.1 *Let H be the operator introduced above and let $\mathbf{f}$ satisfy Assumption 3.1. Then the mapping H is measurable and $\boldsymbol{\theta} \doteq H\mathbf{f}$ is a stationary RCLL process taking values in Ψ.*

For any $t \ge 0$ we define the operator $H_+(t) : \mathcal{S}_{+,0}^N \to \Psi$ in a natural way so that it maps the input flows $f \in \mathcal{S}_{+,0}^N$ *after time* 0 into $H_+(t)f \in \Psi$, the state of the system at time t *given that the system is empty at time* 0. In other words, this mapping is defined by the same expressions (4.10), (4.11) and (4.12), with the additional constraints $t \ge 0$ and $s \ge 0$. As above, it can be also shown that the mapping $H_+(t)$ is measurable for any $t \ge 0$.

4.3 Stationary Maximal Weighted Delay $\hat{\mathbf{r}}^G(0)$

We are now in a position to precisely define the stationary maximal weighted delay $\hat{\mathbf{r}}^G(0)$ that was used in the statement of our main result, Theorem 3.2. Consider a discipline $G \in \mathcal{G}$ and the associated operators $\hat{A}^G$ and $\hat{R}^G$. The stationary maximal weighted delay $\hat{\mathbf{r}}^G = (\hat{\mathbf{r}}^G(t), -\infty < t < \infty)$ is defined by

$$\hat{\mathbf{r}}^G(t) \doteq [\hat{R}^G \mathbf{g}](\mathbf{b}(t)) ,$$

where $(\mathbf{b}(t), \mathbf{g}) = \boldsymbol{\theta}(t) = [H\mathbf{f}](t) \in \Psi$. Note that we do not refer to $\hat{\mathbf{r}}^G$ as a stationary "process" because although $\hat{\mathbf{r}}^G = \hat{R}^G\mathbf{f}$ is a well defined function (and with probability 1 defines the process that corresponds intuitively with our notion of the stationary delay), it *need not be measurable.* If the function is measurable, then $\hat{\mathbf{r}}^G$ is indeed a stationary random process. It can be shown (see [8]) that the operator $\hat{A}$ (and therefore $\hat{R}$) associated with the LWDF discipline is measurable, and thus the probability $P\{\hat{\mathbf{r}}(0)/n > 1\}$ is well defined. The following connection between the operators $\hat{R}^G$ and H_+ is quite straightforward, and will be useful later in the paper. For any $f \in \mathcal{S}_{+,0}^N$ and any $t \ge 0$

$$[\hat{R}^G f](t) = [\hat{R}^G g](b(t)), \tag{4.13}$$

where $(b(t), g) = H_+(t)f$.

5 An "LDP" for the Sequence of Stationary State Processes

In this section we introduce more definitions and then formulate a kind of LDP for the sequence of distributions of $\{\Gamma^n\boldsymbol{\theta}(0), n = 1, \ldots, \}$ in Lemma 5.3. We then briefly indicate how this can be used to obtain the main result Theorem 3.2.

Definition 5.1 *Given any $f \in \mathcal{C}_a^N$ and $r \geq 0$, we define its r-modification f^r to be such that for $i = 1, \ldots, N$,*

$$f_i^r(t) \doteq \begin{cases} \lambda_i t & \text{for } 0 \leq t \leq r, \\ \lambda_i r + f_i(t-r) & \text{for } t > r. \end{cases} \tag{5.14}$$

Definition 5.2 *Given any subset $B \subset \Psi$ and any $s > 0$, we define the set*

$$\Phi_s(B) \doteq \left\{ \begin{array}{c} f \in \mathcal{C}_a^N : \text{ for every } r > 0, \text{ there exists } \delta > 0 \text{ such that} \\ \text{if } h \in U_\delta^{(r+s)}, \text{ then } [H_+(r+s)]h \in B \end{array} \right\}, \tag{5.15}$$

where

$$U_\delta^{(r+s)} \doteq \{h \in \mathcal{S}_{+,0}^N : ||h - f^r||_{r+s} < \delta\}, \tag{5.16}$$

and f^r is as defined in (5.14).

Note that in the above definition $B \subset \Psi$ need not be measurable.

Lemma 5.3 *Let $J_s(f) \doteq \sum_1^N J_s^i(f_i)$. Then the following holds.*

1. *For any measurable subset $B \subset \Psi$,*

$$\limsup_{n\to\infty} \frac{1}{n} \log P\left(\Gamma^n\boldsymbol{\theta}(0) \in B\right) \leq - \inf_{s>0, f\in\overline{H_+^{-1}(s)B}} J_s(f) \ . \tag{5.17}$$

2. *For any (not necessarily measurable) subset $B \subset \Psi$,*

$$\liminf_{n\to\infty} \frac{1}{n} \log P_*\left(\Gamma^n\boldsymbol{\theta}(0) \in B\right) \geq - \inf_{s>0, f\in\Phi_s(B)} J_s(f) \ . \tag{5.18}$$

where P_* denote inner measure with respect to the probability P.

Remark The above statement is not of course a usual LDP. The lower bound is more general than usual large deviations lower bounds due to the use of the inner measure. As mentioned earlier, the reason we need this generality is because we apply this lower bound to the case when the discipline $G \in \mathcal{G}$ is arbitrary, and therefore the associated subset B may be non-measurable.

As shown in Section 6 of [8], using Lemma 5.3, the relation (4.13) with G equal to the LWDF discipline, and certain additional properties, it is possible to show that

$$J_* \doteq \inf_{s>0, f\in\mathcal{C}_a^N : Rf(s)\geq 1} J_s(f), \tag{5.19}$$

satisfies 1) and 2) in Theorem 3.2, where R is an appropriate extension of the operator $\hat{R}$ defined for fluid inputs $f \in \mathcal{C}^N$. Since this extension is rather involved [8, Section 3], for lack of space we cannot define it more precisely here. However, using properties of this operator it is possible to show that the infimum in (5.19) is achieved on a set of "simple" trajectories, and use that to derive the characterization (3.9) of J_* as the solution to a finite-dimensional optimization problem [8, Section 4].

6 Conclusion

In this paper we introduce a new queueing (scheduling) discipline called Largest Weighted Delay First (LWDF) and prove that it is optimal in the sense that it maximizes the asymptotic rate of decay of the tails of the stationary maximal weighted delay within a rather general class $\mathcal{G}$ of queueing disciplines. Even the two restrictions imposed on the class $\mathcal{G}$ appear to be mainly technical. For instance it is only natural to expect any optimal discipline to be work conserving. In addition we also derive a LDP for the maximal weighted stationary delay process associated with LWDF and characterize the rate function in terms of a finite-dimensional optimization problem. The solution of this problem allows one to determine the feasibility of satisfying asymptotic QoS constraints for a given set of users, and hence has significant implications for admission control.

An interesting and very challenging open problem is that of determining optimal disciplines in the network context.

Acknowledgments : We would like to thank our colleagues at Bell Labs M.Andrews, A.Elwalid, K.Kumaran, D.Mitra, M.Reiman, A.Weiss, and P.Whiting for very useful discussions during the course of this work.

References

[1] A.Dembo, O.Zeitouni. Large Deviations Techniques and Applications. Springer, 1998. (2nd edition)

[2] P. Dupuis and R. S. Ellis. *A Weak Convergence Approach to the Theory of Large Deviations.* John Wiley & Sons, New York, 1997.

[3] P. Dupuis and K. Ramanan. A Skorokhod problem formulation and large deviation analysis of a processor sharing model. *Queueing Systems*, 28:109–124, 1998.

[4] R.M.Loynes. The Stability of a Queue with Non-Independent Inter-Arrival and Service Times *Proc. Camb. Phil. Soc.*, 58(3):497-520, 1962.

[5] N. O'Connell. Large deviations for queue lengths at a multi-buffered resource. Technical report HPL-BRIMS-96-010, BRIMS, Hewlett-Packard Labs, Bristol, England, 1996.

[6] I. C. Paschalidis. *Large Deviations in High Speed Communication Networks.* PhD thesis, MIT, Cambridge, Massachusetts, 1996.

[7] A. Shwartz, A. Weiss. *Large Deviations for Performance Analysis: Queues, Communication and Computing.* Chapman and Hall, New York, 1995.

[8] A. L. Stolyar, K. Ramanan. Largest Weighted Delay First Scheduling: Large Deviations and Optimality *Bell Labs Technical Memorandum* BL0112120-990823-25TM, 1999.

On Estimating Buffer Overflow Probabilities under Markov-modulated Inputs*

Ioannis Ch. Paschalidis
Department of Manufacturing Engineering
Boston University
Boston, MA 02215
e-mail: yannisp@bu.edu

Spyridon Vassilaras
Department of ECE
Boston University
Boston, MA 02215
e-mail: svassila@bu.edu

September 1999

Abstract

Based on large deviations asymptotics, we propose estimators for the buffer overflow probability in queues fed by a Markov-modulated input process and serviced by an autocorrelated service process. We demonstrate that these estimators are less likely to underestimate the overflow probability than the estimator obtained by certainty equivalence. As such, they are appropriate in situations where the overflow probability is associated with Quality of Service (QoS) and we need to provide firm QoS guarantees. We also show that as the number of observations increases the proposed estimators converge with probability 1 to the appropriate target and thus do no lead to underutilization of the system in this limit.

Keywords: Large Deviations, Markov-modulated processes, Estimation, Effective Bandwidth, Empirical Measure.

1 Introduction

Estimating small buffer overflow probabilities has found important applications in a variety of application areas. Two prominent examples include the control of *communication networks* and *manufacturing systems.* In modern high-speed communication networks congestion manifests itself as buffer overflows; the *Quality of Service (QoS)* faced by various connections can be quantified by the buffer overflow probability. To provide QoS guarantees the so called *effective bandwidth* admission control mechanism has been proposed (see [10] and references therein). In make-to-stock manufacturing systems, on the other hand, the objective is to control the inventory in order to avoid stockouts (see [1, 6]). In such systems demand is met from a finished goods inventory, and it is backlogged if inventory is not available. The stockout probability quantifies the QoS encountered by customers. It can be shown that this probability is equal to a buffer overflow probability in a corresponding make-to-order system [1]. Thus, the problem of estimating the stockout probability can be transformed into one of estimating a buffer overflow probability.

---

*Research partially supported by the NSF under grants NCR-9706148 and ACI-9873339, and by the Nokia Research Center (Boston).

In both these applications, we are interested in estimating buffer overflow probabilities that are very small. Moreover, arrival and service processes are typically autocorrelated (to model bursty traffic in communication networks, and to accommodate realistic demand scenarios and model failure-prone production facilities in make-to-stock manufacturing systems). As a result, obtaining exact analytic expressions is intractable; it is, therefore, natural to focus on asymptotic regimes. To that end, *large deviations* theory (see [3]) has been an important analytical tool. Large deviations techniques have recently been applied to a variety of problems in telecommunications (see [2, 11] and references therein) and make-to-stock manufacturing systems (see [1]).

All of this large deviations work assumes the detailed knowledge of models for the arrival and service processes. In practice however, such traffic models are not known a priori and have to be estimated from real observations. Measurement-based admission control has received some attention recently [8, 9, 4]. In particular, the authors in [8, 9] consider measurement-based admission control schemes in bufferless multiplexers. They investigate how the overflow probability is affected by estimation errors relying on approximations based on the central limit theorem. The author in [4] extends these results by considering large deviations asymptotics in a multiplexer with nonzero buffer.

In this paper, we will focus on queues fed by Markov-modulated arrival processes. Let Ξ be the transition probability matrix of the underlying Markov-chain of the arrival process and $g(\Xi)$ be the large deviations expression for the overflow probability, when Ξ is known. Let $\hat{\Xi}$ be an unbiased estimator of the transition probability matrix Ξ. It is quite clear that due to the estimation error in Ξ and the nonlinearity of $g(\cdot)$, using $g(\hat{\Xi})$ as an estimator for the overflow probability might result in significant estimation errors. Our goal is to develop new estimates of overflow probabilities that take estimation errors into account. To that end, we will establish an inverse of Sanov's theorem (see [3]) for Markov chains in the large deviations regime, which generalizes a result in [5]. Intuitively, the proposed estimators suggest that we should quote a quantity that is larger than $g(\hat{\Xi})$ to guard against estimation errors. We will provide analytical and numerical evidence that the proposed estimators are "safe" even when based on relatively few observations, meaning that they do not lead to substantial underestimation of the overflow probability that compromise QoS. Still, they are *consistent* in the sense that they converge to $g(\Xi)$ with probability 1 (w.p.1) as the number of observations tends to infinity.

On the organization of the rest of this paper, we start in Section 2 by reviewing some analytical large deviations expressions of buffer overflow probabilities and formulate the estimation problem. In Section 3 we analyze the estimation process in the large deviations regime and develop an inverse of Sanov's theorem for Markov chains. We use this result in Section 4 to propose our estimators of buffer overflow probabilities. Comparisons between various estimators and some illustrative numerical results are in Section 5. Conclusions are in Section 6.

2 Large Deviations of a G/G/1 Queue

In this section, we review work on the large deviations behaviour of a G/G/1 queue under a Markov-modulated arrival process. In particular, we state an asymptotic expression for the probability that the queue length process exceeds a certain threshold U. This probability is the quantity we wish to estimate from measurements.

Consider a single class queue. We will be using a discrete-time model, where time is divided into time slots. We let A_i, $i \in \mathbb{Z}$, denote the aggregate number of customers that enter the queue at time i. The queue has an infinite buffer and is serviced according to a general service process which can clear up to B_i customers during the time interval $[i, i+1]$. We assume that the stochastic processes $\{A_i;\ i \in \mathbb{Z}\}$ and $\{B_i;\ i \in \mathbb{Z}\}$ are stationary, mutually independent, but possibly autocorrelated processes.

For such a process $\{X_i;\ i \in \mathbb{Z}\}$, we define its *limiting log-moment generating function* as the limit

$$\Lambda_X(\theta) \triangleq \lim_{n\to\infty} \frac{1}{n} \log \mathbf{E}[e^{\theta S_n}],$$

where $S_n = \sum_{i=1}^{n} X_i$, provided that this limit exists.

We denote by L_i the queue length at time i (without counting arrivals at time i). We assume that the server uses a work-conserving policy (i.e., the server never stays idle when there is work in the system) and that $\mathbf{E}[A_1] < \mathbf{E}[B_1]$, which by stationarity carries over to all i. We also assume that the queue length process $\{L_i, i \in \mathbb{Z}\}$ is stationary. To simplify the analysis we consider a discrete-time "fluid" model, meaning that we will be treating A_i, L_i, and B_i as nonnegative real numbers.

An LDP for the queue length process has been established in [7] and is given in the next proposition. In preparation for the result, consider a convex function $g(u)$ with the property $g(0) = 0$. We define the *largest root* of $g(u)$ to be the solution of the optimization problem $\sup_{u:g(u)<0} u$.

Proposition 2.1 *The steady-state queue length process L_i satisfies*

$$\lim_{U\to\infty} \frac{1}{U} \log \mathbf{P}[L_i \geq U] = -\theta^*, \tag{1}$$

where $\theta^* > 0$ is the largest root of the equation $\Lambda_A(\theta) + \Lambda_B(-\theta) = 0$.

More intuitively, for large enough U we have $\mathbf{P}[L_i \geq U] \sim e^{-U\theta^*}$. This expression can be used to estimate the overflow probability in a queue with a finite buffer of size U. Kelly [10] establishes that the latter probability has the same asymptotic decay rate (same exponent) with $\mathbf{P}[L_i \geq U]$.

Assume next that the arrival process is Markov-modulated. More specifically, consider an irreducible Markov chain with M states $1, 2, \ldots, M$ and transition probability matrix $\Xi = \{p(i,j)\}_{i,j=1}^{M}$. We will be using the notation $\mathbf{p}' = (\mathbf{p}_1', \ldots, \mathbf{p}_M')$, where $\mathbf{p}_i$ is the ith row of Ξ and prime denotes transpose. The Markov chain makes one transition per time slot; let Y_i be the state at time i. The number of arrivals at time i is a deterministic function of the state, i.e., $A_i = f_A(Y_i)$.

In [3, Sec. 3.1.1] it is established that the limiting log-moment generating function of the arrival process A, denoted by $\Lambda_A(\theta, \mathbf{p})$, is equal to $\log \rho(\mathbf{\Pi}_{\theta,\mathbf{p}}^{A})$, where $\rho(\mathbf{\Pi}_{\theta,\mathbf{p}}^{A})$ denotes the Perron-Frobenius eigenvalue of the $M \times M$ matrix $\mathbf{\Pi}_{\theta,\mathbf{p}}^{A} = \{p(k,j)e^{\theta f_A(j)}\}_{i,j=1}^{M}$. (In this Markov-modulated case we are using notation that explicitly denotes the dependence of $\Lambda_A(\theta, \mathbf{p})$ and $\mathbf{\Pi}_{\theta,\mathbf{p}}^{A}$ on the transition probabilities $\mathbf{p}$.) Notice that because the quantities $e^{\theta f_A(j)}$ are always positive the irreducibility of Ξ implies that $\mathbf{\Pi}_{\theta,\mathbf{p}}^{A}$ is irreducible. This calculation of $\Lambda_A(\theta, \mathbf{p})$ can be easily extended to the case where $f_A(Y_i)$ is a random function of the state Y_i (see [3]).

3 Large Deviations Analysis of the Estimation Process

We start the analysis by studying the large deviations behaviour of the estimation for a Markov-modulated process. In particular, we determine the large deviations rate function of the a posteriori measure for the underlying Markov chain, given the observed empirical measure.

In the simpler case where we are observing the realization of a sequence of i.i.d. random variables, [5] provides the result. We will develop an equivalent of this result in the Markov case. Before we proceed with this agenda, it is instructive to state an equivalent of Sanov's theorem in the Markov case. Consider an irreducible Markov chain with M states $1, 2, \ldots, M$ and transition probability matrix $\Xi = \{p(i,j)\}_{i,j=1}^{M}$. Let $\mathbf{p}$ denote the vector consisting of the rows of Ξ. Let $\mathbf{Y}$ denote a sequence $Y_1, Y_2, \ldots, Y_n$ of states that the Markov chain visits with the initial state being $Y_0 = \sigma$, and consider the empirical measures

$$\mathcal{E}^{\mathbf{Y}}_{n,2}(\mathbf{y}) = \frac{1}{n}\sum_{k=1}^{n} \mathbf{1}_{\mathbf{y}}(Y_{k-1}Y_k),$$

where $\mathbf{y} \in \mathcal{A}^2 \triangleq \{1, \ldots, M\} \times \{1, \ldots, M\}$. Note that when $\mathbf{y} = (i,j) \in \mathcal{A}^2$ the empirical measure $\mathcal{E}^{\mathbf{Y}}_{n,2}(\mathbf{y})$ denotes the fraction of times that the Markov chain makes transitions from i to j in the sequence $\mathbf{Y}$. Let now $\mathcal{A}^2_{\mathbf{p}} \triangleq \{(i,j) \in \mathcal{A}^2 \mid p(i,j) > 0\}$ denote the set of pairs of states that can appear in the sequence $Y_1, Y_2, \ldots, Y_n$ and denote by $M_1(\mathcal{A}^2_{\mathbf{p}})$ the standard $|\mathcal{A}^2_{\mathbf{p}}|$-dimensional probability simplex, where $|\mathcal{A}^2_{\mathbf{p}}|$ denotes the cardinality of $\mathcal{A}^2_{\mathbf{p}}$. Note that the vector of $\mathcal{E}^{\mathbf{Y}}_{n,2}(\mathbf{y})$'s denoted by $\boldsymbol{\mathcal{E}}^{\mathbf{Y}}_{n,2} = (\mathcal{E}^{\mathbf{Y}}_{n,2}(\mathbf{y});\ y \in \mathcal{A}^2_{\mathbf{p}})$ is an element of $M_1(\mathcal{A}^2_{\mathbf{p}})$. For any $\mathbf{q} \in M_1(\mathcal{A}^2_{\mathbf{p}})$, let

$$q_1(i) \triangleq \sum_{j=1}^{M} q(i,j) \qquad \text{and} \qquad q_2(i) = \sum_{j=1}^{M} q(j,i) \tag{2}$$

be its marginals. Whenever $q_1(i) > 0$, let $q_f(j \mid i) \triangleq q(i,j)/q_1(i)$. We will be using the notation $\mathbf{q}_f = (q_f(1 \mid 1), \ldots, q_f(M \mid 1), q_f(1 \mid 2), \ldots, q_f(M \mid 2), \ldots, q_f(1 \mid M), \ldots, q_f(M \mid M))$. We will say that a probability measure $\mathbf{q} \in M_1(\mathcal{A}^2_{\mathbf{p}})$ is *shift invariant* if both its marginals are identical, i.e., $q_1(i) = q_2(i)$ for all i. An LDP for $\boldsymbol{\mathcal{E}}^{\mathbf{Y}}_{n,2}$ is established in the next theorem and is proven in [3, Sec. 3.1.3].

Theorem 3.1 ([3]) *For every* $\mathbf{q} \in M_1(\mathcal{A}^2_{\mathbf{p}})$ *let*

$$I_1(\mathbf{q}) = \begin{cases} \sum_{i=1}^{M} q_1(i) H(q_f(\cdot \mid i) \mid p(i,\cdot)), & \textit{if } \mathbf{q} \textit{ is shift invariant,} \\ \infty, & \textit{otherwise,} \end{cases}$$

where $H(q_f(\cdot \mid i) \mid p(i,\cdot))$ *is the relative entropy, that is,*

$$H(q_f(\cdot \mid i) \mid p(i,\cdot)) = \sum_{j=1}^{M} q_f(j \mid i) \log \frac{q_f(j \mid i)}{p(i,j)}.$$

Then, for any set Γ *of probability vectors in* $M_1(\mathcal{A}^2_{\mathbf{p}})$,

$$-\inf_{\mathbf{q}\in\Gamma^\circ} I_1(\mathbf{q}) \le \lim\inf_{n\to\infty} \frac{1}{n} \log \mathbf{P}[\mathcal{E}^{\mathbf{Y}}_{n,2} \in \Gamma] \le \lim\sup_{n\to\infty} \frac{1}{n} \log \mathbf{P}[\mathcal{E}^{\mathbf{Y}}_{n,2} \in \Gamma] \le -\inf_{\mathbf{q}\in\Gamma} I_1(\mathbf{q}),$$

where Γ° *denotes the interior of* Γ.

We next develop the main result of this section, which can be viewed as the "inverse" of Theorem 3.1. In particular, we observe a certain sequence $\mathbf{Y}$ of states $Y_1, \ldots, Y_n$ in the Markov chain with the initial state being $Y_0 = \sigma$. Let again Ξ the transition probability matrix of the Markov chain and $\mathbf{p}$ the vector of transition probabilities. Note that $\mathbf{p}$ is an element of $(M_1(\mathcal{A}))^M$ (i.e., the M-times cartesian product of $M_1(\mathcal{A})$). We assume that the transition probabilities $\mathbf{p}$ are not known; instead we have a prior $\phi_{\mathbf{p}^-} \in (M_1(M_1(\mathcal{A})))^M$, which assigns probability mass only to $\mathbf{p}$'s corresponding to irreducible Markov chains. Let $\underline{\mathbf{p}}^-$ denote the support of $\phi_{\mathbf{p}^-}$. Given the sequence $\mathbf{Y}$, the posterior distribution is a function of the empirical measure $\mathcal{E}^{\mathbf{Y}}_{n,2}$; we will write $\phi_{\mathbf{p}_n^+(\mathcal{E}^{\mathbf{Y}}_{n,2})}$ for the posterior distribution and $\mathbf{p}_n^+(\mathcal{E}^{\mathbf{Y}}_{n,2})$ for the corresponding random variable.

The main result of this section is the following theorem which is proven in [12].

Theorem 3.2 *Suppose that as* $n \to \infty$ *the empirical measure* $\mathcal{E}^{\mathbf{Y}}_{n,2}$ *converges weekly to some shift invariant* $\mathbf{q}$ *which satisfies* $q_1(i) > 0$, $i = 1, \ldots, M$, *and* $\mathbf{q}_f \in \underline{\mathbf{p}}^-$. *Suppose, also, that* $q(i,j) = 0$ *implies* $\mathcal{E}^{\mathbf{Y}}_{n,2}(i,j) = 0$ *for all* n. *Then for any prior* $\phi_{\mathbf{p}^-}$ *and for every set* Γ *in* $(M_1(\mathcal{A}))^M$ *we have*

$$-\inf_{\mathbf{p}\in\Gamma^\circ} I_2(\mathbf{p}) \le \liminf_{n\to\infty} \frac{1}{n} \log \mathbf{P}[\mathbf{p}_n^+(\mathcal{E}^{\mathbf{Y}}_{n,2}) \in \Gamma] \le \limsup_{n\to\infty} \frac{1}{n} \log \mathbf{P}[\mathbf{p}_n^+(\mathcal{E}^{\mathbf{Y}}_{n,2}) \in \Gamma] \le -\inf_{\mathbf{p}\in\Gamma} I_2(\mathbf{p}),$$

where Γ° *denotes the interior of* Γ *and the rate function is given by*

$$I_2(\mathbf{p}) = \begin{cases} \sum_{i=1}^{M} q_1(i) H(q_f(\cdot \mid i) \mid p(i,\cdot)), & \textit{if } \mathbf{p} \in \underline{\mathbf{p}}^-, \\ \infty, & \textit{otherwise}. \end{cases} \tag{3}$$

4 Estimates of the Overflow Probability

In this section we develop estimates of the overflow probability in a G/G/1 queue and discuss confidence measures.

Consider the model of the G/G/1 queue introduced in Section 2, under a Markov-modulated arrival process. We assume that we have perfect knowledge of the service process B, of the number of states M that characterize the arrival process, and of the function $f_A(\cdot)$ that maps Markov states to amount of arriving fluid. The transition probability matrix Ξ of the underlying Markov chain of the arrival process is unknown. Suppose we observe a sequence $\mathbf{Y} = Y_1, Y_2, \ldots, Y_n$ of states that this Markov chain visits with the initial state being $Y_0 = \sigma$. With the notation of Section 3, and letting the empirical measure $\mathcal{E}^{\mathbf{Y}}_{n,2}$ be equal to $\mathbf{q}_n$, a maximum likelihood estimator of the transition probabilities is given by

$$\hat{p}_n(i,j) = q_{nf}(j \mid i) = \frac{q_n(i,j)}{q_{n1}(i)}, \qquad i,j = 1, \ldots, M, \tag{4}$$

where we assume that n is large enough to have $q_{n1}(i) > 0$ for all i. (To avoid overburdening the notation, in (4) and the sequel we suppress the dependence of the estimators on the sequence $\mathbf{Y}$.) Let $\hat{\mathbf{p}}_n$ denote the vector of these estimates. We can now construct a matrix $\mathbf{\Pi}^A_{\theta,\hat{\mathbf{p}}_n} = \{\hat{p}_n(k,j)e^{\theta f_A(j)}\}_{i,j=1}^M$, and obtain an estimate, $\Lambda_A(\theta, \hat{\mathbf{p}}_n)$, of the limiting log-moment generating function for the arrival process by computing the Perron-Frobenius eigenvalue of $\mathbf{\Pi}^A_{\theta,\hat{\mathbf{p}}_n}$. Applying the result of Proposition 2.1, for large values of U we obtain the following estimate of the overflow probability $\mathbf{P}[L_i \geq U]$

$$\mathcal{P}^I_n \triangleq e^{-U\theta^*(\hat{\mathbf{p}}_n)}, \tag{5}$$

where $\theta^*(\hat{\mathbf{p}}_n)$ is the largest root of the equation $\Lambda_A(\theta, \hat{\mathbf{p}}_n) + \Lambda_B(-\theta) = 0$. If we knew the *true* transition probabilities, say $\mathbf{p}$, then by Proposition 2.1 the ersatz $e^{-U\theta^*(\mathbf{p})}$ approximates the overflow probability; in Eq. (5) we use the estimate $\hat{\mathbf{p}}_n$ in the place of the unknown $\mathbf{p}$. Notice, however, that even though $\hat{\mathbf{p}}_n$ is an unbiased estimate of the transition probabilities, $\mathcal{P}^I_n$ is not necessarily unbiased. Due to the nonlinearity of $e^{-U\theta^*(\cdot)}$, estimation errors in $\hat{\mathbf{p}}_n$ can be severely amplified.

To address this drawback, we next propose an alternative estimator. In particular, taking the Bayesian approach of Section 3 we consider the estimator

$$\mathbf{E}[e^{-U\theta^*(\mathbf{p}^+_n(\boldsymbol{\mathcal{E}}^{\mathbf{Y}}_{n,2}))}], \tag{6}$$

where the expectation is taken with respect to the posterior probability distribution of the transition probabilities. Recall that we have assumed the prior to assign probability mass only to measures corresponding to irreducible Markov chains. As a result, the posterior is also irreducible and $\theta^*(\mathbf{p}^+_n(\boldsymbol{\mathcal{E}}^{\mathbf{Y}}_{n,2}))$ is well defined. Calculating the expectation in (6) exactly is intractable; we will resort to asymptotics. In particular, we let $U = ns$, where s is a scalar, and compute the dominant exponent in the next Proposition which is a direct consequence of Varadhan's integral Lemma [3, Theorem 4.3.1]. For a complete proof see [12].

Proposition 4.1 *It holds*

$$\lim_{n\to\infty} \frac{1}{n} \log \mathbf{E}[e^{-ns\theta^*(\mathbf{p}^+_n(\boldsymbol{\mathcal{E}}^{\mathbf{Y}}_{n,2}))}] = - \inf_{\mathbf{p}\in(M_1(\mathcal{A}))^M} [s\theta^*(\mathbf{p}) + I_2(\mathbf{p})].$$

Note that, as shown in [12] if $\mathbf{p}$ is the vector of transition probabilities corresponding to an irreducible Markov chain, then $\theta^*(\mathbf{p})$ is a continuous function of $\mathbf{p}$.

Based on Eq. (6) and the result of Proposition 4.1, for large values of n we obtain the following estimator of the overflow probability $\mathbf{P}[L_i \geq ns]$

$$\mathcal{P}^{II}_n \triangleq \exp\left\{-n \inf_{\mathbf{p}\in(M_1(\mathcal{A}))^M} [s\theta^*(\mathbf{p}) + I_2(\mathbf{p})]\right\}. \tag{7}$$

A couple of remarks on this estimator are in order. Consider a G/G/1 queue serviced by an independent copy of the service process B, and fed by a Markov-modulated arrival process with transition probabilities distributed according to $\phi_{\mathbf{p}^+_n(\boldsymbol{\mathcal{E}}^{\mathbf{Y}}_{n,2})}$ but independent of the process A. Let $L(\mathbf{p}^+_n(\boldsymbol{\mathcal{E}}^{\mathbf{Y}}_{n,2}))$ denote the queue length in this queue. It can be shown (see [12]) that for large enough n we have

$$\mathbf{P}[L(\mathbf{p}_n^+(\boldsymbol{\mathcal{E}}_{n,2}^{\mathbf{Y}})) > ns] \sim \exp\left\{-n \inf_{\mathbf{p}\in(M_1(\mathcal{A}))^M} [s\theta^*(\mathbf{p}) + I_2(\mathbf{p})]\right\}. \tag{8}$$

This interpretation of $\mathcal{P}_n^{II}$ is similar to the result in [4, Theorem8]; the difference is that [4] does not consider the Markov-modulated case and estimates the limiting log-moment generating function directly from measurements.

To understand on a more intuitive level the estimator $\mathcal{P}_n^{II}$, note that when the empirical measure $\boldsymbol{\mathcal{E}}_{n,2}^{\mathbf{Y}}$ converges to some $\mathbf{q}$, $I_2(\mathbf{q}_f) = 0$. Furthermore, we have assumed in Theorem 3.2 that $\mathbf{q}_f \in \underline{\mathbf{p}}^-$. Consequently,

$$\mathcal{P}_n^{II} = \exp\left\{-n \inf_{\mathbf{r}\in(M_1(\mathcal{A}))^M} [s\theta^*(\mathbf{r}) + I_2(\mathbf{r})]\right\} \geq e^{-ns\theta^*(\mathbf{q}_f)}. \tag{9}$$

The right hand side of the above is the large deviations asymptotic for the overflow probability, if we believe the transition probabilities to be given by $\mathbf{q}_f$ (their estimated values). Instead, we quote $\mathcal{P}_n^{II}$ which is larger. We can say that we pay an "estimation premium" to guard against estimation errors. Taking into account the interpretation in Eq. (8), $e^{-ns\theta^*(\mathbf{q}_f)}$ underestimates the overflow probability, i.e., *certainty equivalence fails.*

If we view the estimator $\mathcal{P}_n^{II}$ as an expectation of the loss measure $e^{-U\theta^*(\mathbf{p}_n^+(\boldsymbol{\mathcal{E}}_{n,2}^{\mathbf{Y}}))}$, we can also compute higher order moments. More specifically, consider the quantities

$$\mathbf{E}[e^{-kU\theta^*(\mathbf{p}_n^+(\boldsymbol{\mathcal{E}}_{n,2}^{\mathbf{Y}}))}], \qquad k = 1, 2, \ldots . \tag{10}$$

Using exactly the same reasoning as in the proof of Proposition 4.1 we can compute the dominant exponent of these quantities. We can thus define estimators based on higher moments of the loss measure at hand. In particular, for $k = 1, 2, \ldots$ we define

$$\mathcal{P}_n^{III}(k) \triangleq \exp\left\{-n \inf_{\mathbf{p}\in(M_1(\mathcal{A}))^M} [ks\theta^*(\mathbf{p}) + I_2(\mathbf{p})]\right\}. \tag{11}$$

Interestingly enough, we can also asymptotically characterize the distribution of the loss measure $e^{-U\theta^*(\mathbf{p}_n^+(\boldsymbol{\mathcal{E}}_{n,2}^{\mathbf{Y}}))}$. The next Proposition describes the result (for a proof see [12]).

Proposition 4.2 *We have*

$$\lim_{n\to\infty} \frac{1}{n} \log \mathbf{P}[e^{-U\theta^*(\mathbf{p}_n^+(\boldsymbol{\mathcal{E}}_{n,2}^{\mathbf{Y}}))} \geq \epsilon] = - \inf_{\substack{\theta^*(\mathbf{p})\leq -\log\epsilon/U \\ \mathbf{p}\in(M_1(\mathcal{A}))^M}} I_2(\mathbf{p}). \tag{12}$$

5 Comparisons and Numerical results

In this section we compare the various estimators discussed so far and present some illustrative numerical results.

The result of Proposition 2.1 suggests approximating the overflow probability $\mathbf{P}[L_i \geq$

$U]$ with $e^{-U\theta^*(\mathbf{p})}$ (see [11, 1] for related approximations and numerical results indicating that this approximation is very accurate for a wide range of U's). We pretend now that we don't know $\mathbf{p}$ and we are interested in estimating the loss probability $\mathbf{P}[L_i \geq U]$ from measurements. In the sequel, when computing various estimators we will be treating the buffer size U as constant; we will discuss the behaviour of the estimators as the number of measurements n increases.

Given that we have observed a sequence $\mathbf{Y}$ of n transitions of the Markov chain corresponding to the arrival process, we will consider three estimators: (i) $\mathcal{P}_n^I$, (ii) $\mathcal{P}_n^{II}$, and (iii) $\mathcal{P}_n^{IV}(m) \triangleq \mathcal{P}_n^{II} + m\sqrt{\mathcal{P}_n^{III}(2) - (\mathcal{P}_n^{II})^2}$, where $\mathcal{P}_n^I$, $\mathcal{P}_n^{II}$, and $\mathcal{P}_n^{III}(\cdot)$ were defined in (5), (7), and (11), respectively, and m is some scalar. Recall that in Section 4 we interpreted $\mathcal{P}_n^{II}$ as the expectation of the loss measure $e^{-U\theta^*(\mathbf{p}_n^+(\mathcal{E}_{n,2}^{\mathbf{Y}}))}$, and $\mathcal{P}_n^{III}(2)$ as the corresponding second moment. In this light, $\mathcal{P}_n^{IV}(m)$ can be interpreted as expectation plus m times standard deviation. To compute these estimators we will be using the empirical measure $\mathcal{E}_{n,2}^{\mathbf{Y}}$ as an approximation of its limit $\mathbf{q}$, which appears in the definition of the rate function $I_2(\cdot)$. Maintaining the notation established so far, we let $\mathcal{E}_{n,2}^{\mathbf{Y}} = \mathbf{q}_n$, and denote by $\mathbf{q}_{nf}$ the corresponding vector of transition probabilities. The next proposition (proven in [12]) compares the three estimators discussed above.

Proposition 5.1 *For any nonnegative n, m it holds:* $\mathcal{P}_n^I \leq \mathcal{P}_n^{II} \leq \mathcal{P}_n^{IV}(m)$. *Furthermore, for all m*

$$\lim_{n\to\infty} \mathcal{P}_n^I = \lim_{n\to\infty} \mathcal{P}_n^{II} = \lim_{n\to\infty} \mathcal{P}_n^{IV}(m) = e^{-U\theta^*(\mathbf{p})}, \qquad w.p.1. \tag{13}$$

Essentially, this proposition establishes that the estimators considered are *consistent* since they converge to the "target" $e^{-U\theta^*(\mathbf{p})}$. We can view $e^{-U\theta^*(\mathbf{p})}$ as the target because this is what we would have quoted as the overflow probability if we knew the true transition probabilities $\mathbf{p}$. When we estimate an overflow probability we are more concerned with underestimating the "true" value than with overestimating it. The reason is that, as outlined in the introduction, the overflow probability quantifies QoS; controlling the queue to guarantee the desired QoS is important. In this light, the estimators $\mathcal{P}_n^{II}$ and $\mathcal{P}_n^{IV}(m)$ are "safer" than $\mathcal{P}_n^I$ since they are less likely to underestimate the overflow probability; yet they are consistent and converge to the correct target $e^{-U\theta^*(\mathbf{p})}$ as $n \to \infty$. The parameter m in $\mathcal{P}_n^{IV}(m)$ determines how conservative we choose to be for relatively small values of n.

We next present a numerical example. We considered a Markov-modulated arrival process, A, driven by a Markov chain with 5 states and a deterninistic service process. The actual values of the parameters used in this experiment can be found in [12]. To improve the accuracy of the large deviations asymptotic we used a refinement discussed in [11] and [1] that introduces a constant in front of the exponential. In particular, the refined large deviations asymptotic is

$$\mathbf{P}[L_i \geq U] \sim \mathcal{P}^{LD} \triangleq \alpha e^{-U\theta^*(\mathbf{p})}, \tag{14}$$

where $a = \mathbf{E}[L_i]\theta^*(\mathbf{p})$ (see [11, 1] for details). In the particular example we are considering, we have $\mathcal{P}^{LD} = 4.9 \times 10^{-7}$. For comparison, simulation yields 6.3×10^{-7}, which indicates that $\mathcal{P}^{LD}$ is a good approximation (in the sense that it captures the order of magnitude and approximates well the first significant digit).

To compare the different estimators of the overflow probability we generated 10,000 long sample paths of the arrival process. For every one of these sample paths and for all $n = 1, 2, \ldots$ we computed a refined version of $\mathcal{P}_n^I$, $\mathcal{P}_n^{II}$ and $\mathcal{P}_n^{IV}(3)$ [1]. We are interested in the fraction of sample paths that lead to a substantial underestimation of $\mathcal{P}^{LD}$. Let $\mathcal{X}_n$ be either $\mathcal{P}_n^I$, $\mathcal{P}_n^{II}$ or $\mathcal{P}_n^{IV}(3)$. In Figure 1, for all such $\mathcal{X}_n$ and n we plot the fraction of sample paths yielding $\mathcal{X}_n$ such that there exists $k \geq n$ with $\mathcal{X}_k < 2.5 \times 10^{-7}$. It can be seen that

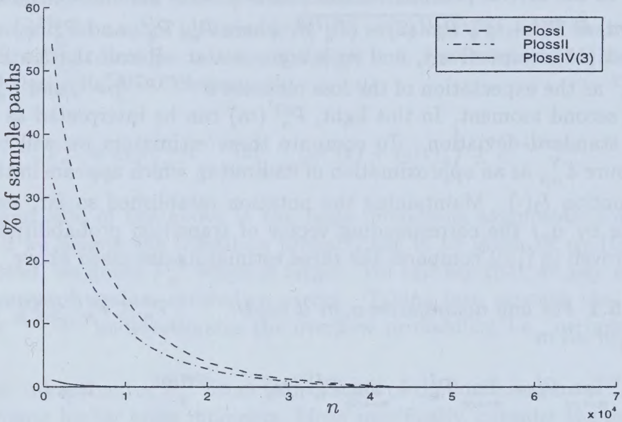


Figure 1: Comparing the three estimators $\mathcal{P}_n^I$, $\mathcal{P}_n^{II}$, and $\mathcal{P}_n^{IV}(3)$. In the legend, "PlossI", "PlossII", and "PlossIV(3)" denotes $\mathcal{P}_n^I$, $\mathcal{P}_n^{II}$, and $\mathcal{P}_n^{IV}(3)$, respectively.

$\mathcal{P}_n^{II}$ is "safer" (in the sense discussed above) than $\mathcal{P}_n^I$, with the difference being significant for small n. Moreover, $\mathcal{P}_n^{IV}(3)$ is "safe" even for very small n (around 2,000). To put the values of n into perspective, note that estimating an overflow probability on the order of 10^{-7} by directly observing the occupancy of the queue (i.e., measuring the fraction of time that we have an overflow) requires more than $O(10^8)$ observations. Of course, to avoid underestimation errors the price to pay is that for small n and for a number of sample paths the estimators $\mathcal{P}_n^{II}$ and $\mathcal{P}_n^{IV}(3)$ overestimate the overflow probability and lead to underutilization of the system. Nevertheless, according to Proposition 5.1 as the number of measurements increases they converge to the appropriate target.

6 Conclusions

We have proposed estimators of buffer overflow probabilities in buffers fed by Markov-modulated inputs. These estimators are based on analytical large deviations asymptotics.

---

[1] To avoid overburdening the notation we will use the same symbols, $\mathcal{P}_n^I$, $\mathcal{P}_n^{II}$ and $\mathcal{P}_n^{IV}(3)$, to denote the refined estimators.

In particular, we considered a situation where the transition probababilities of the Markov chain characterizing the arrival process are not known. For any prior distribution of these transition probababilities we characterized the large deviations behaviour of the posterior distribution. We utilized the latter one to define the proposed estimators. We demonstrated that the proposed estimators are more preferable than the estimator based on certainty equivalence and than observing the occupancy of the queue directly.

References

[1] D. Bertsimas and I. Ch. Paschalidis. Probabilistic service level guarantees in make-to-stock manufacturing systems. Technical report, Department of Manufacturing Engineering, Boston University, February 1999; Revised September 1999. to appear *Operations Research*, available at `http://ionia.bu.edu`.

[2] D. Bertsimas, I. Ch. Paschalidis, and J. N. Tsitsiklis. Asymptotic buffer overflow probabilities in multiclass multiplexers: An optimal control approach. *IEEE Transactions on Automatic Control*, 43(3):315–335, 1998.

[3] A. Dembo and O. Zeitouni. *Large Deviations Techniques and Applications.* Springer-Verlag, NY, 2nd edition, 1998.

[4] N.G. Duffield. A large deviation analysis of errors in measurement based admission control to buffered and bufferless resources. In *Proceedings of the IEEE INFOCOM*, 1999.

[5] A. Ganesh and N. O'Connell. An inverse of Sanov's theorem. Technical report, BRIMS, HP Labs, Bristol, U.K., 1998.

[6] P. Glasserman. Bounds and asymptotics for planning critical safety stocks. *Operations Research*, 45(2):244–257, 1997.

[7] P.W. Glynn and W. Whitt. Logarithmic asymptotics for steady-state tail probabilities in a single-server queue. *J. Appl. Prob.*, 31A:131–156, 1994.

[8] M. Grossglauser and D. Tse. A framework for robust measurement-based admission control. Preprint, 1998.

[9] M. Grossglauser and D. Tse. A time-scale decomposition approach to measurement-based admission control. In *Proceedings of the IEEE INFOCOM*, 1999.

[10] F. P. Kelly. Notes on effective bandwidths. In S. Zachary, I.B. Ziedins, and F.P. Kelly, editors, *Stochastic Networks: Theory and Applications*, volume 9, pages 141–168. Oxford University Press, 1996.

[11] I. Ch. Paschalidis. Class-specific quality of service guarantees in multimedia communication networks. Technical report, Department of Manufacturing Engineering, Boston University, June 1998; Revised 1999. to appear *Automatica* (Special Issue on Control Methods for Communication Networks), Anantharam and Walrand Eds, available at `http://ionia.bu.edu`.

[12] I.Ch. Paschalidis and S. Vassilaras. On the estimation of buffer overflow probabilities from measurements. Technical report, Dept. of Manufacturing Eng., Boston University, 1999.

Induced Burstiness in Generalized Processor Sharing Queues with Long-Tailed Traffic Flows

Sem Borst*,**,†, Onno Boxma*,**, Predrag Jelenković‡

*CWI, P.O. Box 94079, 1090 GB Amsterdam, The Netherlands
**Department of Mathematics & Computing Science, Eindhoven University of Technology
P.O. Box 513, 5600 MB Eindhoven, The Netherlands
†Bell Laboratories, Lucent Technologies, P.O. Box 636, Murray Hill, NJ 07974, USA
‡Department of Electrical Engineering, Columbia University, New York, NY 10027, USA

Abstract

We analyze the asymptotic behavior of long-tailed traffic flows under the Generalized Processor Sharing (GPS) discipline. GPS-based scheduling algorithms, such as Weighted Fair Queueing, have emerged as important mechanisms for achieving differentiated quality-of-service in integrated networks.

We prove that, in certain scenarios, a flow may be strongly affected by the activity of 'heavier'-tailed flows, and may inherit their traffic characteristics, causing induced burstiness. This phenomenon contrasts with previous results which show that, under certain conditions, an individual flow with long-tailed traffic characteristics is effectively served at a constant rate. In particular, the flow is then essentially immune from excessive activity of flows with 'heavier'-tailed traffic characteristics. The stark contrast in qualitative behavior illustrates the crucial importance of the GPS weights in protecting individual flows.

1 Introduction

Statistical data analysis has provided convincing evidence of long-tailed (subexponential [14]) traffic characteristics in high-speed communication networks. Early indications of the long-range dependence of Ethernet traffic, attributed to long-tailed file size distributions, were reported in [19]. Long-tailed characteristics of the scene length distribution of MPEG video streams were explored in [15, 18].

These empirical findings have encouraged active theoretical developments in the modeling and queueing analysis of long-tailed traffic phenomena. We refer to Boxma & Dumas [9] for a comprehensive survey on fluid queues with long-tailed arrival processes. See also Jelenković [16] for an extensive list of references on subexponential queueing models.

Despite significant progress, however, the practical implications are not yet thoroughly understood, in particular issues relating to control and priority mechanisms in the network. To gain a better understanding of those issues, the present paper analyzes the queueing behavior of long-tailed traffic flows under the Generalized Processor Sharing (GPS) discipline. As a design paradigm, GPS is at the heart of commonly-used scheduling algorithms for high-speed switches, such as Weighted Fair Queueing, see for instance Parekh & Gallager [22, 23].

The impact of priority mechanisms on long-tailed traffic phenomena has received relatively little attention. Some recent studies have investigated the effect of the scheduling discipline on the waiting-time distribution in the classical M/G/1 queue, see for instance Anantharam [2]. For FCFS, it is well-known [12] that the waiting-time tail is regularly varying of index $1 - \nu$ iff the service time tail is regularly varying of index $-\nu$. For LCFS preemptive resume as well as for Processor Sharing, the waiting-time tail turns out to be regularly varying of the *same* index as the service time tail [8], [26], although with different pre-factors. In the case of Processor Sharing with several customer classes, Zwart [24] recently showed that the sojourn time distribution of a class-i customer is regularly varying of index $-\nu_i$ iff the service time distribution of that class is regularly varying of index $-\nu_i$, *regardless* of the service time distributions of the other classes. In contrast, for two customer classes with ordinary non-preemptive priority, the tail behavior of the waiting- and sojourn time distributions is determined by the *heaviest* of the (regularly-varying) service time distributions [1].

In the present paper, we consider the Generalized Processor Sharing (GPS) discipline. GPS-based scheduling algorithms, such as Weighted Fair Queueing, have emerged as important mechanisms for achieving differentiated quality-of-service in integrated-services networks. The queueing analysis of GPS is extremely difficult. Interesting partial results for exponential traffic models were obtained in Dupuis & Ramanan [13] and Massoulié [20] among others.

Here, we focus on non-exponential traffic models. We show that, in certain scenarios, a flow may be strongly affected by the activity of 'heavier'-tailed flows, and may inherit their traffic characteristics, causing induced burstiness. This complements results previously obtained in [5] which show that, under certain conditions, an individual flow with long-tailed traffic characteristics is effectively served at a constant rate. The latter rate only depends on the traffic characteristics of other flows through their average rate. In particular, the flow is then largely insensitive to extreme activity of flows with 'heavier'-tailed traffic characteristics. The stark contrast in qualitative behavior underscores the crucial importance of the GPS weights in protecting individual flows.

The remainder of the paper is organized as follows. In Section 2, we present a detailed model description. In Section 3, we derive generic lower and upper bounds for the buffer content distribution. We then show, in Section 4, that for long-tailed traffic characteristics, the lower and upper bounds have the same asymptotic behavior, yielding exact asymptotics for the tail distribution of the buffer content.

2 Model description

Consider two flows sharing a link of unit rate. Traffic from the flows is served in accordance with the Generalized Processor Sharing (GPS) discipline, which operates as follows. There are weights ϕ_i, $i = 1, 2$, associated with each of the flows, with $\phi_1 + \phi_2 = 1$. As long as both flows are backlogged, flow i is served at rate ϕ_i, $i = 1, 2$. If one of the flows is not backlogged, however, then the capacity is reallocated to the other flow, which is then served at the full link rate (if backlogged).

We ignore some technicalities here which may arise for general arrival processes when the inflow rate r_i of source i may be smaller than the weight ϕ_i. In that case, only the *excess* capacity, i.e., $\phi_i - r_i$, is reallocated to the other flow. These subtleties however will not arise for the arrival processes that we consider. We refer to Dupuis & Ramanan [13] for a more thorough discussion of these issues.

Denote by $V_i(t)$ the buffer content of flow i at time t. Let $\mathbf{V}_i$ be a stochastic variable with as distribution the limiting distribution of $V_i(t)$ for $t \to \infty$ (assuming it exists). Denote by $A_i(s,t)$ the amount of work generated by flow i during the time interval $(s,t]$, and assume that A_i is a stationary process. Define $B_i(s,t)$ as the amount of service received by flow i during $(s,t]$. Then the following identity relation holds for all $0 \leq s \leq t$:

$$V_i(t) = V_i(s) + A_i(s,t) - B_i(s,t). \tag{1}$$

For any $c \geq 0$, denote by $V_i^c(t) := \sup\limits_{0 \leq s \leq t} \{A_i(s,t) - c(t-s)\}$ the workload at time t in a queue with constant service rate c fed by the arrival process of flow i. Denote by ρ_i the traffic intensity of flow i. For $c > \rho_i$, let $\mathbf{V}_i^c$ be a stochastic variable with as distribution the limiting distribution of $V_i^c(t)$ for $t \to \infty$. Also, define $P_i^c(\cdot)$ as the distribution function of the associated busy period, and denote by $\mathbf{P}_i^c$ a stochastic variable with that distribution. For conciseness, we will occasionally suppress the superscript c when the service rate is clear from the context.
Analogously to (1), the following identity relation holds for all $0 \leq s \leq t$:

$$V_i^c(t) = V_i^c(s) + A_i(s,t) - B_i^c(s,t), \tag{2}$$

with

$$B_i^c(s,t) = c \int_s^t \mathrm{I}_{\{V_i^c(u)>0\}} \mathrm{d}u. \tag{3}$$

Before describing the traffic model, we first introduce some further notation. For any two real functions $g(\cdot)$ and $h(\cdot)$, we use the notational convention $g(x) \sim h(x)$ to denote $\lim_{x\to\infty} g(x)/h(x) = 1$, or equivalently, $g(x) = h(x)(1 + o(1))$ as $x \to \infty$. For any stochastic variable $\mathbf{X}$ with distribution function $F(\cdot)$, $\mathbb{E}\mathbf{X} < \infty$, denote by $F^r(\cdot)$ the distribution function of the residual lifetime of $\mathbf{X}$, i.e., $F^r(x) = \frac{1}{\mathbb{E}\mathbf{X}} \int_0^x (1 - F(y))\mathrm{d}y$, and by $\mathbf{X}^r$ a stochastic variable with that distribution.
The classes of *long-tailed*, *subexponential*, *regularly varying*, *intermediately regularly varying*, and *dominatedly varying* distributions are denoted with the symbols $\mathcal{L}$, $\mathcal{S}$, $\mathcal{R}$, $\mathcal{IR}$, and $\mathcal{DR}$, respectively. The definitions of these classes may be found in references [3], [11], [14]. We note that the following inclusion relation holds: $\mathcal{R} \subset \mathcal{IR} \subset (\mathcal{DR} \cap \mathcal{L}) \subset \mathcal{S} \subset \mathcal{L}$.

We now describe the two traffic scenarios that we consider.

2.1 Instantaneous arrivals

Here, a source generates instantaneous traffic bursts according to a renewal processes. The interarrival times between bursts of source i have distribution function $U_i(\cdot)$ with mean $1/\lambda_i$. The burst sizes of source i have distribution $S_i(\cdot)$ with mean σ_i. Thus, the traffic intensity of source i is $\rho_i = \lambda_i \sigma_i$.

We now state two results which will play a crucial role in the analysis.

Theorem 2.1 *(Pakes [21]) If $S_i^r(\cdot) \in \mathcal{S}$, and $\rho_i < c$, then*

$$\mathbb{P}\{\mathbf{V}_i^c > x\} \sim \frac{\rho_i}{c - \rho_i} \mathbb{P}\{\mathbf{S}_i^r > x\}.$$

Theorem 2.2 *(Zwart [25]) If $S_i(\cdot) \in \mathcal{IR}$, and $\rho_i < c$, then*

$$\mathbb{P}\{\mathbf{P}_i > x\} \sim \frac{c}{c-\rho_i}\mathbb{P}\{\mathbf{S}_i > x(c-\rho_i)\}.$$

In the analysis we will need a slight modification of the preceding theorem:

Theorem 2.3 *If $U_i(\cdot)$ is an exponential distribution, i.e., the arrival process is Poisson, and $S_i^r(\cdot) \in \mathcal{IR}$, then*

$$\mathbb{P}\{\mathbf{P}_i^r > x\} \sim \frac{c}{c-\rho_i}\mathbb{P}\{\mathbf{S}_i^r > x(c-\rho_i)\}.$$

Remark 2.1 *The above theorem can be proved by mimicking the proof of Theorem 2.6 (see Remark 2.2 below). Note that if $S_i(\cdot) \in \mathcal{IR}$, then Theorem 2.2 implies Theorem 2.3. However, if we only assume $S_i^r(\cdot) \in \mathcal{IR}$, then we cannot directly use Theorem 2.2, since $S_i^r(\cdot) \in \mathcal{IR}$ does not necessarily imply $S_i(\cdot) \in \mathcal{IR}$.*

2.2 On-Off processes

Here, a source generates traffic according to an On-Off process, alternating between On- and Off-periods. The Off-periods of source i have distribution function $U_i(\cdot)$ with mean $1/\lambda_i$. The On-periods of source i have distribution $S_i(\cdot)$ with mean σ_i. While On, source i produces traffic at a constant rate r_i, so the mean burst size is $\sigma_i r_i$. The fraction of time that source i is Off is

$$p_i = \frac{1/\lambda_i}{1/\lambda_i + \sigma_i} = \frac{1}{1+\lambda_i\sigma_i}.$$

The traffic intensity of source i is

$$\rho_i = (1-p_i)r_i = \frac{\lambda_i\sigma_i r_i}{1+\lambda_i\sigma_i}.$$

We now state the analogues of Theorems 2.1 and 2.2 for the case of On-Off processes.

Theorem 2.4 *(Jelenković & Lazar [17]) If $S_i^r(\cdot) \in \mathcal{S}$, and $\rho_i < c < r_i$, then*

$$\mathbb{P}\{\mathbf{V}_i^c > x\} \sim p_i\frac{\rho_i}{c-\rho_i}\mathbb{P}\{\mathbf{S}_i^r > x/(r_i-c)\}.$$

Theorem 2.5 *(Boxma & Dumas [10]) If $S_i(\cdot) \in \mathcal{R}$, and $\rho_i < c < r_i$, then*

$$\mathbb{P}\{\mathbf{P}_i > x\} \sim p_i\frac{c}{c-\rho_i}\mathbb{P}\{\mathbf{S}_i > x(c-\rho_i)/(r_i-\rho_i)\}.$$

In addition, the following minor extension of the preceding theorem holds:

Theorem 2.6 *If $U_i(\cdot)$ is an exponential distribution, $S_i^r(\cdot) \in \mathcal{IR}$, and $\rho_i < c < r_i$, then*

$$\mathbb{P}\{\mathbf{P}_i^r > x\} \sim p_i\frac{c}{c-\rho_i}\mathbb{P}\{\mathbf{S}_i^r > x(c-\rho_i)/(r_i-\rho_i)\}.$$

Remark 2.2 *Although the above theorem is only a minor extension of Theorem 2.5, the proof is new and might be of independent interest. It directly uses Theorem 2.4 to derive the asymptotic behavior of the residual busy period. Theorem 2.3 can be proved in a similar fashion. The only difference is that one uses Theorem 2.1 instead of Theorem 2.4.*

Proof of Theorem 2.6

Due to space limitations, the proof is omitted, and may be found in [7]. The proof is not essential for the understanding of the main theme of the paper.

◇

3 Bounds

We now derive some generic bounds for the buffer content which we will use in the next section to analyze its tail distribution. Without loss of generality we focus on flow 1. The bounds apply for the scenario of instantaneous arrivals as well as On-Off processes as described in Subsections 2.1 and 2.2, respectively.
We first introduce some notation. Define

$$\begin{aligned} Q^{\epsilon}(t) &:= \sup_{0 \leq s \leq t} \left\{ (\phi_2 + \rho_1 + \epsilon - 1)(t-s) - \phi_2 \int_s^t \mathrm{I}_{\{V^{\phi_2}(u)=0\}} du \right\} \\ &= \sup_{0 \leq s \leq t} \{ B_2^{\phi_2}(s,t) - (1 - \rho_1 - \epsilon)(t-s) \}, \end{aligned}$$

with $B_2^{\phi_2}(s,t)$ as in (3). For $0 < \epsilon < 1 - \rho_1 - \rho_2$, let $\mathbf{Q}^{\epsilon}$ be a stochastic variable with as distribution the limiting distribution of $Q^{\epsilon}(t)$ for $t \to \infty$. Define $R_1^c(s) := \sup_{u \geq s}\{c(u-s) - A_1(s,u)\}$. For $c < \rho_1$, let $\mathbf{R}_1^c$ be a stochastic variable with as distribution the distribution of $R_1^c(s)$ (which does not depend on s since the process A_1 is stationary).

We first present a lower bound for the buffer content distribution of flow 1.

Lemma 3.1 *(Lower bound) For any $\epsilon > 0$ sufficiently small and y,*

$$\mathbb{P}\{\mathbf{V}_1 > x\} \geq \mathbb{P}\{\mathbf{Q}^{-\epsilon} > x + y\} \, \mathbb{P}\{\mathbf{R}_1^{\rho_1 - \epsilon} \leq y\}.$$

Proof
Define

$$s^* := \arg \sup_{0 \leq s \leq t} \{ B_2^{\phi_2}(s,t) - (1 - \rho_1 + \epsilon)(t-s) \}, \tag{4}$$

so that

$$Q^{-\epsilon}(t) = B_2^{\phi_2}(s^*,t) - (1 - \rho_1 + \epsilon)(t - s^*). \tag{5}$$

It is easily verified that for any $\epsilon > 0$ sufficiently small,

$$V_2^{\phi_2}(s^*) = 0, \tag{6}$$

because otherwise s^* would not be the maximizing argument in (4), as $\phi_2 = 1 - \phi_1 > 1 - \rho_1 + \epsilon$.
The GPS discipline ensures that

$$V_2(t) \leq V_2^{\phi_2}(t), \tag{7}$$

since each flow i is guaranteed to receive a minimum service rate ϕ_i whenever backlogged. Combining (1), (2), (6), and (7),

$$B_2(s^*,t) \geq B_2^{\phi_2}(s^*,t). \tag{8}$$

By definition,

$$B_1(s,t) + B_2(s,t) \leq t - s,$$

for all $0 \leq s \leq t$, so that

$$B_1(s^*,t) \leq t - s^* - B_2(s^*,t). \tag{9}$$

Substituting (8), (9) into (2), with $i = 1$, using (5),

$$\begin{aligned}
V_1(t) &\geq A_1(s^*,t) + B_2^{\phi_2}(s^*,t) - (t - s^*) \\
&= Q^{-\epsilon}(t) + A_1(s^*,t) - (\rho_1 - \epsilon)(t - s^*) \\
&\geq Q^{-\epsilon}(t) + \inf_{u \geq s^*} \{A_1(s^*,u) - (\rho_1 - \epsilon)(u - s^*)\} \\
&= Q^{-\epsilon}(t) - \sup_{u \geq s^*} \{(\rho_1 - \epsilon)(u - s^*) - A_1(s^*,u)\} \\
&= Q^{-\epsilon}(t) - R_1^{\rho_1 - \epsilon}(s^*).
\end{aligned}$$

Note that, by (4), (5), s^*, $Q^{-\epsilon}(t)$ depend only on $A_2(s,t), 0 \leq s \leq t$, and are independent of $R_1^{\rho_1-\epsilon}(s), s \geq 0$ (fixed). Hence,

$$\begin{aligned}
\mathbb{P}\left\{V_1(t) > x | s^*\right\} &\geq \mathbb{P}\left\{Q^{-\epsilon}(t) > x - y, R_1^{\rho_1-\epsilon}(s^*) \leq y | s^*\right\} \\
&= \mathbb{P}\left\{Q^{-\epsilon}(t) > x - y | s^*\right\} \mathbb{P}\left\{\mathbf{R}_1^{\rho_1-\epsilon} \leq y\right\},
\end{aligned}$$

which immediately yields the statement of the lemma.

◇

We now present an upper bound for the buffer content distribution of flow 1.

Lemma 3.2 *(Upper bound) For any $\epsilon > 0$ and δ,*

$$\mathbb{P}\left\{\mathbf{V}_1 > x\right\} \leq \mathbb{P}\left\{\mathbf{Q}^{\epsilon} > (1 - \delta)x\right\} + \mathbb{P}\left\{\mathbf{V}_1^{\rho_1+\epsilon} > \delta x\right\}.$$

Proof
The GPS discipline implies that

$$V_1(t) = \sup_{0 \leq s \leq t} \{A_1(s,t) - C_1(s,t)\}, \tag{10}$$

with

$$C_1(s,t) = \int_s^t (1 - \phi_2 \mathrm{I}_{\{V_2(u) > 0\}}) du. \tag{11}$$

From (3), (7),

$$C_1(s,t) = \int_s^t (1 - \phi_2 \mathrm{I}_{\{V_2(u) > 0\}}) du \geq \int_s^t (1 - \phi_2 \mathrm{I}_{\{V_2^{\phi_2}(u) > 0\}}) du = t - s - B_2^{\phi_2}(s,t). \tag{12}$$

Substituting (11), (12) into (10), for any $\epsilon > 0$,

$$\begin{aligned}
V_1(t) &\leq \sup_{0 \leq s \leq t} \{A_1(s,t) - (t - s) + B_2^{\phi_2}(s,t)\} \\
&\leq \sup_{0 \leq s \leq t} \{A_1(s,t) - (\rho_1 + \epsilon)(t - s)\} + \sup_{0 \leq s \leq t} \{B_2^{\phi_2}(s,t) - (1 - \rho_1 - \epsilon)(t - s)\} \\
&= V_1^{\rho_1+\epsilon}(t) + Q^{\epsilon}(t),
\end{aligned}$$

from which the statement of the lemma directly follows.

◇

4 Asymptotic analysis

We now use the bounds from the previous section to determine the tail distribution of the buffer content. We consider both the scenario of instantaneous arrivals and On-Off processes as described in Subsections 2.1 and 2.2, respectively. We also allow for mixed traffic scenarios, where one flow generates instantaneous arrivals, while the other produces traffic according to an On-Off process.
We now state the main theorem of the paper. For conciseness, denote $\mathbf{P}_2^r := (\mathbf{P}_2^{\phi_2})^r$.

Theorem 4.1 *Assume that* $U_2(\cdot)$ *is an exponential distribution.*
If $\rho_1 > \phi_1$, $\rho_1 + \rho_2 < 1$ *(which implies* $\rho_2 < \phi_2$*),* $r_2 > \phi_2$, $S_2^r(\cdot) \in \mathcal{IR}$, *and* $\mathbb{P}\{\mathbf{S}_1^r > x\} = o(\mathbb{P}\{\mathbf{S}_2^r > x\})$ *as* $x \to \infty$, *then*

$$\mathbb{P}\{\mathbf{V}_1 > x\} \sim \frac{\phi_2 - \rho_2}{\phi_2} \frac{\rho_2}{1 - \rho_1 - \rho_2} \mathbb{P}\{\mathbf{P}_2^r > x/(\rho_1 - \phi_1)\},$$

with $\mathbb{P}\{\mathbf{P}_2^r > x/(\rho_1 - \phi_1)\}$ *as in Theorems 2.3 and 2.6, respectively.*

The above theorem complements the results obtained in [5] which in the context of the present model may be formulated as follows.

Theorem 4.2 *If either (i)* $\rho_1 < \phi_1$, *and* $S_1^r(\cdot) \in \mathcal{S}$ *(instantaneous arrivals) or* $S_1^r(\cdot) \in \mathcal{IR}$ *(On-Off processes); or (ii)* $S_1^r(\cdot) \in \mathcal{IR}$, $\mathbb{P}\{\mathbf{S}_2^r > x\} = o(\mathbb{P}\{\mathbf{S}_1^r > x\})$ *as* $x \to \infty$, *and* $r_1 > c_1$ *in case of On-Off processes, then*

$$\mathbb{P}\{\mathbf{V}_1 > x\} \sim \mathbb{P}\{\mathbf{V}_1^{c_1} > x\},$$

with $c_1 := \max\{\phi_1, 1 - \rho_2\}$ *and* $\mathbb{P}\{\mathbf{V}_1^{c_1} > x\}$ *as in Theorems 2.1 and 2.4, respectively.*

Before giving the formal proof of Theorem 4.1, we first provide an intuitive interpretation. When flow 2 is backlogged, flow 1 is only served at rate ϕ_1, while it generates traffic at average rate $\rho_1 > \phi_1$. Thus the queue of flow 1 has positive drift $\rho_1 - \phi_1 > 0$ when flow 2 is backlogged. Now suppose that flow 2 generates a large burst or experiences a long On-period. It will then become backlogged, and because of the positive drift, flow 1 will soon become backlogged too; flow 2 will thus experience a busy period as if it were served at rate ϕ_2; flow 1 will be served at rate ϕ_1, and its queue will roughly grow at rate $\rho_1 - \phi_1$.
Of course, its queue may also build up when flow 1 itself generates a large burst or experiences a long On-period. However, these effects are dominated by the build-up during the busy periods of flow 2, because the traffic characteristics of flow 2 are heavier. Thus, the most likely scenario for flow 1 to build a large queue is for flow 2 to generate a large burst, or experience a long On-period, while flow 1 itself shows average behavior. As long as flow 2 is backlogged, the queue of flow 1 will then roughly grow at rate $\rho_1 - \phi_1$. When flow 2 is not backlogged, the queue of flow 1 will drain at approximately rate $1 - \rho_1$. Thus, the queue of flow 1 behaves as a queue with constant service rate $1 - \rho_1$ fed by an On-Off process, with On- and -Off periods equal to the busy and idle periods of flow 2 when served at constant rate ϕ_2. This is reflected in Theorem 4.1, if we use Theorem 2.4 to interpret the right-hand side.
In contrast, under the assumptions of Theorem 4.2, the above scenario cannot arise: either (i) $\rho_1 < \phi_1$, so that the queue of flow 1 retains negative drift when flow 2 is backlogged; or (ii) $\mathbb{P}\{\mathbf{S}_2^r > x\} = o(\mathbb{P}\{\mathbf{S}_1^r > x\})$ as $x \to \infty$ so that the congestion effects

due to activity of flow 1 itself dominate the build-up during the busy-periods of flow 2. In this case, the most likely scenario for flow 1 to build a large queue is to generate a large burst or experience a long On-period itself, while flow 2 shows average behavior. Flow 1 will then approximately be served at a constant rate c_1, as confirmed by Theorem 4.2. In [6], analogues of Theorems 4.1 and 4.2 were obtained for a closely related coupled-processors model using an explicit expression for the workload transforms. In addition, the analysis in [6] covers the theoretically interesting case that $\mathbb{P}\{\mathbf{S}_1 > x\} \sim K\mathbb{P}\{\mathbf{S}_2 > x\}$ with $0 < K < \infty$.

We now give the proof of Theorem 4.1.

Proof of Theorem 4.1
First observe that $Q^\epsilon(t)$ represents the workload at time t in a queue with constant service rate $c = 1 - \rho_1 - \epsilon$ fed by an On-Off process with On- and Off-periods equal to the busy and idle periods associated with the workload process $V^{\phi_2}(t)$, respectively. During the On-periods, traffic is produced at constant rate $r = \phi_2$. The fraction Off-time is $p = 1 - \rho_2/\phi_2$. The On- and Off-periods are independent because $U_2(\cdot)$ is an exponential distribution. As in the proof of Theorem 2.6 in [7], it may be shown that $S_2^r(\cdot) \in \mathcal{S}$ implies $P_2^r(\cdot) \in \mathcal{S}$. Hence, from Theorem 2.4,

$$\mathbb{P}\{\mathbf{Q}^\epsilon > x\} \sim \frac{\phi_2 - \rho_2}{\phi_2} \frac{\rho_2}{1 - \rho_1 - \rho_2 - \epsilon} \mathbb{P}\{\mathbf{P}_2^r > x/(\rho_1 + \epsilon - \phi_1)\}. \tag{13}$$

From Theorems 2.3 and 2.6, for $0 < \alpha < 1$,

$$\limsup_{x\to\infty} \frac{\mathbb{P}\{\mathbf{P}_2^r > \alpha x\}}{\mathbb{P}\{\mathbf{P}_2^r > x\}} = K(\alpha), \tag{14}$$

with $\lim_{\alpha \uparrow 1} K(\alpha) = 1$; an analogous statement holds for $\alpha > 1$ with lim inf instead of lim sup and $\lim_{\alpha \downarrow 1} K(\alpha) = 1$.
(Lower bound) Denote $d_1 = \rho_1 - \phi_1$. Using (13), (14), and Lemma 3.1,

$$\begin{aligned}
\liminf_{x\to\infty} \frac{\mathbb{P}\{\mathbf{V}_1 > x\}}{\mathbb{P}\{\mathbf{P}_2^r > x/d_1\}} &\geq \mathbb{P}\{\mathbf{R}_1^{\rho_1+\epsilon} \leq y\} \liminf_{x\to\infty} \frac{\mathbb{P}\{\mathbf{Q}^{-\epsilon} > x + y\}}{\mathbb{P}\{\mathbf{P}_2^r > x/d_1\}} \\
&= \mathbb{P}\{\mathbf{R}_1^{\rho_1+\epsilon} \leq y\} \frac{\phi_2 - \rho_2}{\phi_2} \frac{\rho_2}{1 - \rho_1 - \rho_2 - \epsilon} \\
&\qquad \liminf_{x\to\infty} \frac{\mathbb{P}\{\mathbf{P}_2^r > (x+y)/(d_1 - \epsilon)\}}{\mathbb{P}\{\mathbf{P}_2^r > x/d_1\}} \\
&= \frac{\phi_2 - \rho_2}{\phi_2} \frac{\rho_2}{1 - \rho_1 - \rho_2 - \epsilon} K\left(\frac{d_1}{d_1 - \epsilon}\right) \mathbb{P}\{\mathbf{R}_1^{\rho_1+\epsilon} \leq y\}.
\end{aligned}$$

Letting $y \to \infty$, then $\epsilon \downarrow 0$,

$$\liminf_{x\to\infty} \frac{\mathbb{P}\{\mathbf{V}_1 > x\}}{\mathbb{P}\{\mathbf{P}_2^r > x/(\rho_1 - \phi_1)\}} \geq \frac{\phi_2 - \rho_2}{\phi_2} \frac{\rho_2}{1 - \rho_1 - \rho_2}.$$

(Upper bound) Using Theorems 2.1, 2.3, 2.4, and 2.6, it may be verified that

$$\mathbb{P}\{\mathbf{V}_1^{\rho_1+\epsilon} > \delta x\} = o(\mathbb{P}\{\mathbf{P}_2^r > x/(\rho_1 - \phi_1)\}) \quad \text{as} \quad x \to \infty. \tag{15}$$

Using (13), (14), (15), and Lemma 3.2,

$$\begin{aligned}
\limsup_{x\to\infty} \frac{\mathbb{P}\{\mathbf{V}_1 > x\}}{\mathbb{P}\{\mathbf{P}_2^r > x/d_1\}} &\leq \limsup_{x\to\infty} \frac{\mathbb{P}\{\mathbf{Q}^\epsilon > (1-\delta)x\}}{\mathbb{P}\{\mathbf{P}_2^r > x/d_1\}} + \limsup_{x\to\infty} \frac{\mathbb{P}\{\mathbf{V}_1^{\rho_1+\epsilon} > \delta x\}}{\mathbb{P}\{\mathbf{P}_2^r > x/d_1\}} \\
&= \frac{\phi_2-\rho_2}{\phi_2}\frac{\rho_2}{1-\rho_1-\rho_2+\epsilon} \limsup_{x\to\infty} \frac{\mathbb{P}\{\mathbf{P}_2^r > (1-\delta)x/(d_1+\epsilon)\}}{\mathbb{P}\{\mathbf{P}_2^r > x/d_1\}} \\
&= \frac{\phi_2-\rho_2}{\phi_2}\frac{\rho_2}{1-\rho_1-\rho_2+\epsilon} K\left(\frac{(1-\delta)d_1}{d_1+\epsilon}\right).
\end{aligned}$$

Letting $\delta, \epsilon \downarrow 0$,

$$\limsup_{x\to\infty} \frac{\mathbb{P}\{\mathbf{V}_1 > x\}}{\mathbb{P}\{\mathbf{P}_2^r > x/(\rho_1-\phi_1)\}} \leq \frac{\phi_2-\rho_2}{\phi_2}\frac{\rho_2}{1-\rho_1-\rho_2}.$$

◇

References

[1] Abate, J., Whitt, W. (1997). Asymptotics for M/G/1 low-priority waiting-time tail probabilities. *Queueing Systems* **25**, 173-233.

[2] Anantharam, V. (1997). Scheduling strategies and long-range dependence. Technical report, University of California, Berkeley.

[3] Bingham, N.H., Goldie, C.M., Teugels, J.L. (1987). *Regular Variation* (Cambridge University Press, Cambridge, UK).

[4] Borst, S.C., Boxma, O.J., Jelenković, P.R. (1999). Generalized Processor Sharing with long-tailed traffic sources. In: *Teletraffic Engineering in a Competitive World, Proc. ITC-16*, Edinburgh, UK, eds. P. Key, D. Smith (North-Holland, Amsterdam), 345-354.

[5] Borst, S.C., Boxma, O.J., Jelenković, P.R. (1999). Asymptotic behavior of Generalized Processor Sharing with long-tailed traffic sources. Submitted for publication.

[6] Borst, S.C., Boxma, O.J., Jelenković, P.R. (1999). Coupled processors with regularly varying service times. Technical Report COSOR 99-11 Eindhoven University of Technology. Submitted for publication.

[7] Borst, S.C., Boxma, O.J., Jelenković, P.R. (1999). Induced burstiness in generalized processor sharing queues with long-tailed traffic flows. Technical Report.
`URL: http://comet.ctr.columbia.edu/~predrag`

[8] Boxma, O.J., Cohen, J.W. (1998). The single server queue: Heavy tails and heavy traffic. In: *Self-similar Network Traffic and Performance Evaluation*, eds. K. Park, W. Willinger (Wiley, New York), to appear.

[9] Boxma, O.J., Dumas, V. (1998). Fluid queues with heavy-tailed activity period distributions. *Computer Communications* **21**, 1509-1529.

[10] Boxma, O.J., Dumas, V. (1998). The busy period in the fluid queue. *Perf. Eval. Review* **26**, 100-110.

[11] Cline, D.B.H. (1994). Intermediate regular and π variation. In: *Proc. London Math. Soc.* **68**, 594-616.

[12] Cohen, J.W. (1973). Some results on regular variation for distributions in queueing and fluctuation theory. *J. Appl. Prob.* **10**, 343-353.

[13] Dupuis, P., Ramanan, K. (1998). A Skorokhod Problem formulation and large deviation analysis of a processor sharing model. *Queueing Systems* **28**, 109-124.

[14] Goldie, C.M., Klüppelberg, C. (1997). Subexponential distributions. In: *A Practical Guide to Heavy Tails: Statistical Techniques and Applications*, eds. R.J. Adler, R.E Feldman, M.S. Taqqu (Birkhäuser Verlag, Basel), 435-459.

[15] Heyman, D.P, Lakshman, T.V. (1996). Source models for VBR broadcast-video traffic. IEEE/ACM *Trans. Netw.* **4**, 40-48.

[16] Jelenković, P.R. (1998). Asymptotic results for queues with subexponential arrival processes. In: *Self-similar Network Traffic and Performance Evaluation*, eds. K. Park, W. Willinger (Wiley, New York), to appear.

[17] Jelenković, P.R., Lazar., A.A. (1996). Multiplexing on-off sources with subexponential on periods. *Adv. Appl. Prob.* **31**, in press.

[18] Jelenković, P.R., Lazar, A.A., Semret, N. (1997). The effect of multiple time scales and subexponentiality of MPEG video streams on queueing behavior. IEEE *J. Sel. Areas Commun.* **15**, 1052-1071.

[19] Leland, W.E., Taqqu, M.S., Willinger, W., Wilson, D.V. (1993). On the self-similar nature of Ethernet traffic. In: *Proc. SIGCOMM '93*, 183-193.

[20] Massoulié, L. (1998). Large deviations for polling and Weighted Fair Queueing service systems. Report France Télécom-CNET.

[21] Pakes, A.G. (1975). On the tails of waiting-time distributions. *J. Appl. Prob.* **12**, 555-564.

[22] Parekh, A.K., Gallager, R.G. (1993). A generalized processor sharing approach to flow control in integrated services networks: the single-node case. IEEE/ACM *Trans. Netw.* **1**, 344-357.

[23] Parekh, A.K., Gallager, R.G. (1994). A generalized processor sharing approach to flow control in integrated services networks: the multiple node case. IEEE/ACM *Trans. Netw.* **2**, 137-150.

[24] Zwart, A.P. (1998). Sojourn times in a multiclass processor sharing queue. In: *Teletraffic Engineering in a Competitive World, Proc. ITC-16*, Edinburgh, UK, eds. P. Key, D. Smith (North-Holland, Amsterdam), 335-344.

[25] Zwart, A.P. (1999). Tail asymptotics for the busy period in the GI/G/1 queue. Technical Report COSOR 99-12 Eindhoven University of Technology. Submitted for publication.

[26] Zwart, A.P., Boxma, O.J. (1998). Sojourn time asymptotics in the M/G/1 processor sharing queue. *Queueing Systems*, to appear.

The Asymptotics of Selecting the Shortest of Two, Improved

Michael Mitzenmacher* Berhold Vöcking †

We investigate variations of a novel, recently proposed load balancing scheme based on small amounts of choice. To motivate this work, we first provide the relevant background. When n balls are thrown into n bins, the maximum load, or balls in a bin, is $\frac{\ln n}{\ln \ln n}(1 + o(1))$ with high probability. Suppose instead that n balls are sequentially placed into n bins so that for each ball, $d \geq 2$ bins are chosen independently and uniformly at random from the n bins, and the ball is placed in the bin with the fewest balls, ties being broken arbitrarily. Then in this case the maximum load is only $\frac{\ln \ln n}{\ln d} \pm \Theta(1)$ with high probability [1].

This result was generalized to natural queueing models independently in [6] and [2, 3]. Suppose that tasks arrive at a bank of n First In First Out processors as a Poisson process of rate λn, where $\lambda < 1$; tasks require an exponentially distributed amount of service with mean 1. If each task queues at a random processor, then in the stationary distribution the probability that a server has at least k tasks is simply λ^k. If instead each task chooses two processors at random and queues at the shorter, then in the fluid limit process representing the limiting behavior as n grows, the probability that a server has at least k tasks converges to $\lambda^{\frac{d^k-1}{d-1}}$. That is, the tails of the processor queue lengths decrease *doubly exponentially*, rather than exponentially, when $d > 1$. The effect is clear in simulations even for relatively small values of n.

In this paper, we consider a variation based on the work by Vöcking [5]. In his basic model, n bins are split into d groups of size n/d. We think of these groups as being laid out linearly, with the first group (bins 1 to n/d) being the *leftmost*, and so on. A ball is placed by choosing a bin independently and uniformly at random from each of the d groups and placing the ball into the least loaded of the d bins; ties are broken by placing the ball in the leftmost (tied) bin. Vöcking showed that the maximum number of balls in a bin in this case is $\frac{\ln \ln n}{d \ln \phi_d} \pm \Theta(1)$ with high probability, where here ϕ_d corresponds to the exponent of growth for a generalized Fibonacci sequence. (For reference, $\phi_2 = 1.61 < \phi_3 < \phi_4 \ldots < 2$.) Surprisingly, coordinating ties improves performance! These results also hold for certain dynamic models; see [5] for more details. To differentiate the approaches, we call Vöcking's the *d-left* scheme and the original the *d-random* scheme.

Here, following the work of [2], we examine the d-left scheme using fluid limit models corresponding to a family of differential equations. Because of limited space we consider only the queueing variation. For more details and results, see the extended draft of this paper [4].

Suppose arriving tasks choose a processor according to the d-random scheme. We let w_i represent the fraction of the bins with at least i tasks in the queue (including the one being served). Then the differential equations describing the fluid limit are

$$\frac{dw_i}{dt} = \lambda(w_{i-1}^d - w_i^d) - (w_i - w_{i+1}). \quad (1)$$

Of course $w_0(t) = 1$ always.

The key to understanding these systems generally revolves around their *fixed points*, where all the derivatives are zero. Moreover, as we expect the system to be stable, we seek fixed points where the w_i fall to 0 as i grows. It can be checked [3, 6] that the unique such fixed point for the d-random queueing system, which we denote by w_i^*, is given by $w_i^* = \lambda^{\frac{d^i-1}{d-1}}$. Hence, we find that even for $d = 2$, the tails of the queue lengths decrease doubly exponentially.

We now examine the d-left scheme. We first consider the case $d = 2$, for which the fluid limit equations are

$$\frac{dy_i}{dt} = 4\lambda\,(y_{i-1} - y_i)\, z_{i-1} - (y_i - y_{i+1}) \;; \quad (2)$$

---

*Harvard University, Computer Science Department. 29 Oxford St., Cambridge, MA 02138. e-mail:michaelm@eecs.harvard.edu.

†International Computer Science Institute, Berkeley, CA 94704. e-mail: voecking@icsi.berkeley.edu. Supported by a grant of the "Gemeinsames Hochschulsonderprogramm III von Bund und Ländern" through the DAAD.

$$\frac{dz_i}{dt} = 4\lambda\left(z_{i-1} - z_i\right) y_i - \left(z_i - z_{i+1}\right), \qquad (3)$$

where y_i and z_i represent the fraction of servers with queue length at least i on the left and right, respectively. We may simplify for general d by using x_{jd+k} to represent the fraction of servers that have queue length at least j and lie in the kth group from the left. Then the fluid limit model yields the following family of differential equations:

$$\frac{dx_i}{dt} = d^d\lambda\left(x_{i-d} - x_i\right)\left(\prod_{j=i-d+1}^{i-1} x_j\right) - \left(x_i - x_{i+d}\right).$$

This system appears substantially more complex than (1), even in the simplest case where $d = 2$. Indeed, we have not been able to prove that there is a unique fixed point (satisfying $x_i \to 0$ as $i \to \infty$), although that appears to be the case based on simulating the process determined by the differential equations. We will from here on assume that there is indeed such a unique fixed point for this system. With this assumption, we can prove that the tails for this system at the fixed point are term by term no greater than the tails at the fixed point of the d-random system. Let $u_i = \sum_{k=0}^{d-1} x_{id+k}$. That is, u_i is the total fraction of servers with load at least i. Clearly $u_0 = 1$. We represent the values at the fixed point for the d-left scheme by u_i^* and x_i^*. Since all derivatives are zero at the fixed point, we have

$$d^d\lambda\left(x_{i-d}^* - x_i^*\right)\left(\prod_{j=i-d+1}^{i-1} x_j^*\right) - \left(x_i^* - x_{i+d}^*\right) = 0.$$

Summing these expressions from $i = jd$ to infinity yields $u_j^* = d^d\lambda\prod_{i=(j-1)d}^{jd-1} x_i^*$. Hence, for instance, it follows that $u_1^* = \lambda$, verifying that the arrival rate equals the departure rate at the fixed point. But now by the arithmetic-geometric mean inequality

$$u_j^* = d^d\lambda\prod_{i=(j-1)d}^{jd-1} x_i^* \leq \lambda\left(\sum_{i=(j-1)d}^{jd-1} x_i^*\right)^d = \lambda\left(u_{j-1}^*\right)^d.$$

A simple direct induction now yields that $u_j^* \leq \lambda^{\frac{d^j-1}{d-1}}$, and hence in equilibrium in the fluid limit the d-random system has larger tails than a d-left system. In fact, as explained in [4], the tails decrease *Fibonacci exponentially*, which is faster than the double exponetial decrease of the d-random system.

	$d = 2$		$d = 3$	
i	w_i^*	u_i^*	w_i^*	u_i^*
1	9.0e-1	9.0e-1	9.0e-1	9.0e-1
2	7.3e-1	7.3e-1	6.5e-1	6.5e-1
3	4.8e-1	4.8e-1	2.5e-1	2.5e-1
4	2.1e-1	2.0e-1	1.4e-2	9.1e-3
5	3.8e-2	3.2e-2	2.9e-6	5.8e-10
6	1.3e-3	5.8e-4	2.9e-17	1.4e-54
7	1.5e-6	2.5e-8	9.7e-51	

Table 1: Results for d-left (u^*) vs. d-random (w^*) in the fluid limit model when $\lambda = 0.9$.

We present results obtained from finding the fixed point from the fluid limit model through simulating the differential equations. We demonstrate the behavior in Table 1 by focusing on the case $\lambda = 0.9$ and comparing fixed point values for the d-left and d-random schemes. The improvement from using the d-left scheme on measures such as the expected time is small, but real. The improvement is more pronounced if we consider a different measure of performance. As we have shown, the d-left scheme leads to a faster decrease in the tails of the loads. Hence, the main benefit of the d-left over the d-random scheme is that the probability of joining a long queue, and thereby spending a long time in the system, is much smaller under the d-left scheme.

References

[1] Y. Azar, A. Broder, A. Karlin, and E. Upfal. Balanced Allocations. In *Proc. of the 26th ACM Symp. on the Theory of Computing*, 1994, pp. 593–602.

[2] M. Mitzenmacher. The Power of Two Choices in Randomized Load Balancing. Ph.D. thesis, U. C. Berkeley, September 1996.

[3] M. Mitzenmacher. Load Balancing and Density Dependent Jump Markov Processes. In *Proc. of the 37th IEEE Symp. on Foundations of Computer Science*, 1996, pp. 213–222.

[4] M. Mitzenmacher and B. Vöcking. The Asymptotics of Selecting the Shortest of Two, Improved. Extended abstract available at www.eecs.harvard.edu/~michaelm/NEWWORK/papers.html.

[5] B. Vöcking, How Asymmetry Helps Load Balancing. Extended abstract, to appear in FOCS '99.

[6] N.D. Vvedenskaya, R.L. Dobrushin, and F.I. Karpelevich. Queueing System with Selection of the Shortest of Two Queues: an Asymptotic Approach. *Problems of Information Transmission*, Vol 32, 1996, pp. 15–27.

Universal Filtering and Prediction of Individual Sequences Corrupted By Noise*

Anelia Baruch
anelia@tx.technion.ac.il
Neri Merhav
merhav@ee.technion.ac.il
Dept. of Electrical Engineering,Technion - IIT, Haifa 32000, Israel

Abstract

We address the problem of estimating the current outcome or predicting the next outcome of an arbitrary individual sequence over a finite alphabet, when only *noisy* past observations are available. The performance criterion investigated is the expected fraction of errors, and the performance of our proposed algorithm is compared to the minimum expected fraction of errors that can be made by any Finite State Machine (FSM). We prove that Markovian machines compete with the entire class of FSMs in the sense of achieving essentially the performance of any FSM. It is also shown that there exists a sequential algorithm that attains the same asymptotic performance as the optimal, non-sequential Markovian scheme. Finally, a scheme that is based on the incremental parsing algorithm of Lempel and Ziv is introduced, that increases the memory length at an appropriate rate, and asymptotically achieves optimality with respect to all FSMs.

1 Introduction

Many problems in Information Theory are aimed at devising universal sequential decision algorithms, that when applied to individual data sequences, attain asymptotically optimum performance with respect to some performance criterion, e.g., mean square error, probability of error, the log-loss function, or Hamming distance. The rate at which a sequential scheme approaches optimality, as well as the amount of resources required, are the main features of interest. Consider the following problem: Let $\theta_1, \theta_2, \ldots$ be an arbitrary deterministic sequence over a finite alphabet, symbolizing the clean signal to be estimated. There are no assumptions regarding its empirical statistics and no underlying probabilistic model is assumed. The noise sequence, $\nu_1, \nu_2, \ldots$, is a sequence of binary i.i.d. r.v.'s, where $Pr\{\nu_i = 1\} = 1 - Pr\{\nu_i = 0\} \stackrel{\triangle}{=} p$. An observer receives sequentially a noisy version of the deterministic sequence $y_1, y_2, \ldots$, where $y_i = \theta_i \oplus \nu_i$ and $\oplus$ denotes addition modulo 2 (XOR), and is interested in either estimating θ_i, or predicting θ_{i+1}, based on $y_1, y_2, \ldots, y_i$. We seek a universal estimator whose performance (in terms of bit error probability) is essentially as good as that of the best FSM, simultaneously for all θ. A FSM is defined [1] as a mechanism in which the decision is a function of the present state, and the observed data recursively determines the transition between finitely many

---

*This research was supported by the Israeli Science Foundation administered by the Israeli Academy of Sciences and Humanity.

states. The finite number of states accounts for the finite amount of resources and storage limitation of any practical algorithm.

This work can be regarded as an extension of the *sequential compound decision problem*, which was introduced by Robbins [2] (see for example, also [3] -[6]), to the case of competing with a general S-state machine rather than $S = 1$: The observer has access to a noisy version of the individual sequence emitted from a memoryless stationary channel, and is interested in sequential estimation as well as the best single state machine in the presence of noise. The minimum expected loss (with respect to some loss function) achievable by an observer who knows the first order empirical statistics of the *clean* individual sequence *in advance*, is referred to as *Bayes Envelope*. For all sufficiently large sequence length n, decision algorithms whose risks are uniformly close to Bayes envelope are found in [3] -[6]. Hannan [4] proved that in the finite alphabet and finite strategy space case, the convergence rate of the performance of the best sequential strategy to the Bayes envelope is upper bounded by $O(n^{-1/2})$. In the continuous case, with a loss function that satisfies some smoothness condition quantified by a parameter α, the convergence rate is $O(1/n \sum_{i=1}^{n} t^{-\alpha})$, which is $O(\log n/n)$ for $\alpha = 1$.

The *k-th extended compound sequential decision problem* ([7], [8]) is an extension of the sequential compound decision problem, in the sense that the decision strategy $\hat{\theta}$ is allowed to be a function of k noisy past observations (k-th order Markovian decision mechanisms). In [7] and [8], smoothness conditions are imposed on the channel, and the problem treated in our work does not fall within these smoothing conditions.

This work can also be regarded as an extension of [1]. Feder, Merhav, and Gutman [1] discussed a scenario in which there is complete feedback, i.e., the observations are noiseless, and the prediction mechanism is an FSM. Loosely speaking, the Finite State (FS) predictability of an infinite sequence is defined as the minimum fraction of prediction errors that can be made by any FS predictor. A sequential scheme that is based on the incremental parsing algorithm of Lempel and Ziv [9] is introduced in [1]. This scheme serves, in fact, as a Markovian machine of increasing order, and is proven to asymptotically attain the FS predictability, universally for all individual sequences.

In a later work, Merhav and Feder [10] studied a wider family of sequential decision problems and additive performance criteria. They provided a sufficient condition on the loss function, which is a version of Lipschitz condition, for the sequential procedure to perform within $O\left(n^{-1} \ln n\right)$ as well as the best fixed strategy. This result holds for finite alphabets as well as infinite ones. Unfortunately, this result does not apply to discontinuous loss functions, and thus precludes the Hamming distance loss function. Therefore, in order to sequentially approach optimality, when the loss functions do not fall within the smoothness condition, randomization is included in the sequential scheme and the convergence rate, in this case, slows down to $O\left(n^{-1/2}\right)$.

Another, completely different approach to the problem of universal prediction, is referred to as *prediction using expert advice* (see for example [11]). Instead of competing with a set of specified strategies, it is assumed that there are N "experts" and there is no limitation whatsoever on the their strategies. Under some regularity conditions on the loss function, an iterative weighting algorithm, that assigns iteratively-calculated weights to each of the experts, is shown to perform asymptotically as well as the best expert, at a rate that depends on the loss function.

One can use the same sequential scheme to compete with a specified class of "experts" such as S-state FSMs, by operating all machines in parallel, and by calculating the loss encountered by each one of them, which is an exhaustive, yet implementable procedure.

In this paper we extend the work [12]. For the sake of completeness, previously attained results are presented and are followed by the description of the Lempel-Ziv based algorithm (Section 5). In Section 2, we define the *FS filterability* of and infinite sequence as the expected minimum fraction of prediction or estimation errors that can be made by any FS machine which is fed by a corrupted version of the clean sequence where the expectation is taken over the ensemble of noise sequences.

The main difficulties in competing with all S-state FSMs are that the minimizing FSM with respect to a given individual sequence is unknown, and that its performance should be attained sequentially, with incomplete feedback (the noisy observations).

The first difficulty is overcome by introducing a sub-class of FSMs, the family of Markovian machines. A k-th order Markovian machine is an FSM whose current state is defined by the last k noisy observations. Markovian machines seem plausible under the reasonable assumption that the near past is more relevant than the remote past. Markovian machines are also very easy to implement. Moreover, motivated by [1], we derive in Section 3 a uniform upper bound on the difference in performance between the best k-th order Markovian machine and the optimal S-state FSM. We also introduce sequential decision rules that compete with the best non-sequential ones (Section 4). We conclude by introducing two sequential Markovian schemes whose order increases as time elapses (Section 5). The first one is referred to as the segmented scheme and the second is based on the incremental parsing algorithm of Lempel and Ziv. The main obstacle in deriving the second algorithm is that the random observations determine the structure of the Markovian scheme and therefore the set of states that are visited is random. Both schemes are shown to attain the FS filterability for every individual sequence. It should be noted that in this paper, most proofs are only outlined or omitted for the sake of brevity, for the detailed derivations see [13].

2 Notation and Definitions

Define a finite-state predictor [1] by the pair (f, g):

$$\hat{\theta}_{i+1} = f(s_i) \tag{1}$$

$$s_{i+1} = g(s_i, y_i), \tag{2}$$

where $s_i \in \mathcal{S}$ is the state at time instant i, $\mathcal{S} \triangleq \{1, \ldots, S\}$ is the set of states whose cardinality is $|\mathcal{S}| = S < \infty$, and $\hat{\theta}_i \in \{0, 1\}$ is the predicted value of θ_i. When a FS predictor (f, g) is fed by an infinite sequence $y_1, y_2, \ldots$, it emits a sequence of estimates $\hat{\theta}_1, \hat{\theta}_2, \ldots$ according to (1) while going through a sequence of states $s_1, s_2, \ldots$, according to (2). When estimation of the current bit is concerned, we allow the decision at time index i to be a function of y_i as well: $\hat{\theta}_i = f(s_i, y_i)$, $s_{i+1} = g(s_i, y_i)$. Henceforth, unless otherwise specified, we refer to a FSM as a general term for either a FS predictor or a FS estimator of the current bit, and $M(S)$ stands for the set of all possible S-state FSMs.

In order to simplify the notation, we shall denote by $x_{n_1}^{n_2}$ $(n_1 \leq n_2)$ the sequence $x_{n_1}, x_{n_1+1}, \ldots, x_{n_2}$.

Define $e\left(\theta_1^n, \nu_1^n, m\right)$ as the recurrence of errors achieved by $m = (f, g) \in M(S)$ when used on a noisy version of the individual sequence y_1^n such that $y_i = \theta_i \oplus \nu_i$. Define the *filterability* of θ_1^n using $(f, g) \in M(S)$ by: $e(\theta_1^n, m) \triangleq E\left\{e\left(\theta_1^n, \nu_1^n, (f, g)\right)\right\}$. Define the *filterability* of θ_1^n using the next state function g as $e_g(\theta_1^n) \triangleq \min_{f \in \mathcal{F}_g} e(\theta_1^n, (f, g))$, where $\mathcal{F}_g$ is the set of all possible decision rules that can be assigned to the states of g. Further, define the *S-state filterability* of θ_1^n by: $e_S(\theta_1^n) \triangleq \min_{g \in G_S} e_g(\theta_1^n), = \min_{m \in M(S)} e(\theta_1^n, m)$, where G_S is the class of all possible S-state next-state functions. Due to the arbitrary nature of θ_1^n, the sequence $\{e_S(\theta_1^n)\}_{n=1}^{\infty}$ does not necessarily converge, and the limit superior expresses a worst-case approach. The *asymptotic S-state filterability* of the sequence $\theta = (\theta_1, \theta_2 \ldots)$ is defined by: $e_S(\theta) \triangleq \limsup_{n \to \infty} e_S(\theta_1^n)$. Finally, define the *FS filterability* of the infinite sequence θ as $e(\theta) = \lim_{s \to \infty} e_S(\theta)$, where the limit exists by the monotonicity of $e_S(\theta)$ (since $\forall S, M(S-1) \subset M(S)$).

The following subsets of $M(S)$ are of special interest:
The class of Markovian machines of order k, $G(2^k) \triangleq \{m \in M(S) : s_i \equiv y_{i-k}^{i-1}\}$, a class of extended Markovian machines, $\tilde{G}(2^k, T) \triangleq \{m \in M(S) : s_i \equiv (y_{i-k}^{i-1}, c_i), c_i = i - \lfloor \frac{i}{T} \rfloor T\}$, and aperiodic machines $\bar{M}(S) \triangleq \{m \in M(S) : g \text{ aperiodic }\}$.

Define the *asymptotic aperiodic Finite State filterability* of an infinite sequence θ : $\bar{e}(\theta) \triangleq \lim_{S \to \infty} \limsup_{n \to \infty} \min_{m \in \bar{M}(S)} Ee(\theta_1^n, \nu_1^n, m)$. Define the *asymptotic Markovian filterability* of an infinite sequence θ : $\mu(\theta) \triangleq \lim_{k \to \infty} \limsup_{n \to \infty} \min_{m \in G(2^k)} Ee(\theta_1^n, \nu_1^n, m)$. Define the *extended asymptotic Markovian filterability* of an infinite sequence θ : $\mu'(\theta) \triangleq \lim_{S \to \infty} \lim_{k \to \infty} \limsup_{n \to \infty} \min_{m \in \tilde{G}(2^k, S!)} e(\theta_1^n, \nu_1^n, m)$.

3 Markovian Machines Versus FSMs

In this section, we relate the performance of the best k-th order Markovian machine with respect to a given sequence θ_1^n, to its filterability to that of the best S-state machine of a given sequence θ_1^n in terms of an inequality of the form $\min_{m_k \in G(2^k)} Ee(\theta_1^n, \nu_1^n, m_k) \leq e_S(\theta_1^n) + \delta(n, S, k)$, where, for a fixed S, $\lim_{k \to \infty} \lim_{n \to \infty} \delta(n, S, k) = 0$. We establish two alternative proofs for two different bounds. The first bound refers to a binary symmetric channel and it holds only for prediction with $\lim_{n \to \infty} \delta(n, S, k) = \frac{1}{1-2p}\sqrt{\frac{\ln S}{2(k+1)}}$. The proof is partially based on the proof of the equivalent claim in [1] for the case of prediction based on clean data. The second bound refers to a more general channel (binary Markovian channel) and to any decision strategy, which is defined by a deterministic function of the current state with $\lim_{n \to \infty} \delta(n, S, k) = K(s, p)\,(\alpha(S, p))^k$, where $K(S, p)$ and $|\alpha(S, p)| < 1$ are constants that depend solely on S and p. The main drawback of the second proof is that periodic machines require special treatment.

3.1 Prediction in Binary Symmetric Channel

The following theorem states that Markovian machines are asymptotically as good as general FSMs when the prediction problem is concerned, and the observations are the output of a binary symmetric channel (BSC) fed by an individual sequence.

Let g_k be the k-th order Markovian next-state function.

Theorem 1. *For prediction, $\forall \theta$, $\forall S < \infty$, and $\forall k$,*

$$\limsup_{n \to \infty} e_{g_k}(\theta_1^n) - e_S(\theta) \leq \frac{1}{1-2p}\sqrt{\frac{\ln S}{2(k+1)}} \tag{3}$$

hence, for prediction, we have $\mu(\theta) = e(\theta)$.

Outline of the Proof of Theorem 1: The proof is based on a comparison of the performance of the best FSM whose aim is to predict at time instant t the value of the next clean outcome, θ_{t+1}, and the performance of the best FSM whose aim is to predict the value of the next corrupted outcome y_{t+1}.

Define $R_g^\theta(\theta_1^n) \triangleq \frac{1}{n}\sum_{s=1}^S \min\left\{EN_n^\theta(s,1), EN_n^\theta(s,0)\right\}$, where $N_n^\theta(s,x)$ is the r.v. designating the joint count of visits to state s and the value of the next *clean* bit being x. It is easy to show that $e_g(\theta_1^n) = R_g^\theta(\theta_1^n)$. It should be noted that if a genie were to tell us $\left\{EN_n^\theta(s,x), x = 0,1\right\}_{s=1}^S$ in advance, $e_g(\theta_1^n)$ could be achieved by the prediction rule

$$\hat{\theta}_{t+1} = \begin{cases} 1 & EN_n^\theta(s_t,1) > EN_n^\theta(s_t,0) \\ 0 & EN_n^\theta(s_t,1) < EN_n^\theta(s_t,0) \\ arbitrary & EN_n^\theta(s_t,1) = EN_n^\theta(s_t,0) \end{cases} \tag{4}$$

Analogously, define $R_g^y(\theta_1^n) \triangleq \frac{1}{n}\sum_{s=1}^S \min\{EN_n^y(s,1), EN_n^y(s,0)\}$, where $N_n^y(s,x)$ is the r.v. designating the joint count of visits to state s and the value of the next *corrupted* bit being x. Note that $R_g^y(\theta_1^n)$ can be viewed as the minimum expected count of errors achieved by any prediction mechanism whose aim is to predict y_{t+1} and is of the form: $s_{t+1} = g(y_t, s_t)$, $\hat{y}_{t+1} = f(s_t)$

Also define a similar quantity by replacing the order of the expectation and minimization operators: $\tilde{R}_g^y(\theta_1^n) \triangleq \frac{1}{n}\sum_{s=1}^S E\min\{N_n^y(s,0), N_n^y(s,1)\}$ that would be achievable if the r.v.'s $\{N_n^y(s,0), N_n^y(s,1)\}_{s=1}^S$ were known in advance.

The proof of the theorem is established by the following arguments: For all θ_1^n, $R_g^\theta(\theta_1^n)$ is expressed as a deterministic function of $R_g^y(\theta_1^n)$: $R_g^\theta(\theta_1^n) = \frac{1}{1-2p}R_g^y(\theta_1^n) - \frac{p}{1-2p}$, and since this relation holds for all next-state functions (including g_k), we get $R_{g_k}^\theta(\theta_1^n) - R_g^\theta(\theta_1^n) = \frac{1}{1-2p}\left(R_{g_k}^y(\theta_1^n) - R_g^y(\theta_1^n)\right)$. Then we derive two upper bounds $R_{g_k}^y(\theta_1^n) - \tilde{R}_{g_k}^y(\theta_1^n) < \sqrt{\frac{\ln S}{2(k+1)}}$ and $\tilde{R}_{g_k}^y(\theta_1^n) - R_g^y(\theta_1^n) < O\left(\frac{2^k(1-p)^{k/2}k^{1/2}}{n^{1/2}}\right)$, where the former is obtained similarly to Theorem 2 in [1], and the latter follows from the fact that the difference $R_{g_k}^y(\theta_1^n) - \tilde{R}_{g_k}^y(\theta_1^n)$ decays asymptotically since the r.v. $N_n^y(y^k,1)$ converges to its expectation, and thus $e_{g_k}(\theta_1^n) - e_S(\theta_1^n) \le \delta_1(n,S,k) = O(2^k k^{1/2}(1-p)^k n^{-1/2}) + O(\sqrt{\ln S/k})$.

In the noise-free case [1], the equivalent bound of $\delta_1(n,S,k)$ is $O(\sqrt{\ln S/k}) + k/n$ (the term k/n is due to end-effects). It is argued that the bound $\delta_1(n,S,k)$ is not tight in general. This claim is supported by Theorem 2 to follow, where a tighter bound which behaves like $O(\alpha(p,S)^k) + k/n$ is obtained. We emphasize that the term $\frac{1}{1-2p}$ in (3) can lead to the wrong conclusion that the convergence of $e_{g_k}(\theta_1^n)$ to $e_S(\theta_1^n)$ slows down as the probability of error (p) increases, where in fact, the converse is true, since as the noise intensity increases, the effect of past states on the current state decreases, and the memory length required to compete with the S-state machine decreases as well.

3.2 Markovian Machines - The Probabilistic Approach

In this section, we state an alternative proof of the claim that one can search for the finite state predictor/estimator in the class of Markovian machines and achieve the same performance. The alternative proof relies on the fact that the state sequence is a Markov chain, and it is more general, in the sense that it can be extended to a more general channel (Markovian channel), and it holds for the prediction problem $\hat{\theta}_{t+1} = f(s_t)$ as well as the estimation of the current bit $\hat{\theta}_t = f(s_t, y_t)$. In fact, the derivation holds for any decision of the form $f(s_t)$. Furthermore, the convergence rate derived in the following proof is exponential with k, while the convergence rate established in the first bound is $O(k^{-1/2})$. The main drawback of the proof that is given in this section, is that periodic FSMs require special analysis, and as a result, the convergence rate deteriorates.

3.2.1 Binary Symmetric Channel

Let an aperiodic FSM be given. A Markovian machine which asymptotically (with the Markovian order) attains the same performance is constructed based on the given FSM. The same construction can be applied to the optimal S-state machine with regard to θ_1^n and thus there exists Markovian machine that achieves $e_S(\theta_1^n)$ (up to ϵ).

Theorem 2. *For $p > 0$,* [1] *and every $\theta \in \{0,1\}_1^\infty$ there exists a Markovian machine $m_{k,n} \in G(2^k)$ satisfying:*

$$e(\theta_1^n, m_{k,n}) \leq \min_{m \in \bar{M}(S)} e(\theta_1^n, m) + K(S,p)\left(\alpha(S,p)\right)^k + \frac{k}{n}. \tag{5}$$

where $|\alpha(S,p)| < 1$ and $K(S,p)$ are functions of S and p that are independent of θ_1^n.

Hence, $\mu(\theta) = \bar{e}(\theta)$.

Outline of the Proof of Theorem 2: First, one observes that transient states of a FSM do not affect the asymptotic performance, so it suffices to compete with irreducible FSMs whose states are all recurrent. Further, if g is irreducible with no transient states, and is aperiodic, then the effect of the initial state on the probability distribution of the states fades away exponentially fast. The derivation of this claim relies on known results concerning non-homogeneous Markov chains (see [14] ch. 3). Finally, a Markovian machine that competes with a given FSM (f,g) is constructed: If one wants to approach $e_g(\theta_1^n)$ within ϵ, one chooses k such that $K(S,p)\left(\alpha(S,p)\right)^k + \frac{k}{n} < \epsilon$. Then, one uses the Markovian machine of order k whose prediction of a bit at state y_{t-k+1}^t, is set to be the same as the decision made by (g,f) with initial state chosen arbitrarily and fed by y_{t-k+1}^t. Since the effect of the initial state fades away exponentially fast, the performance of the constructed Markovian machine approaches $e(\theta_1^n, (g,f))$ within ϵ. Theorem 2 is proved with $K(S,p) = -2S^3 \log p$, and $\alpha(S,p) = \left(\frac{1-p^{S^2}}{1+p^{S^2}}\right)^{1/S^2}$

3.2.2 Binary Markovian Channel

The extension of Theorem 2 to a more general channel model is quite trivial, and it results in a slower convergence. The noise $\nu_1, \nu_2, \ldots$ is assumed to be a Markovian process of a known order M, i.e., M is the minimal integer for which $\forall \{\alpha_i\}_{i=1}^n \in \{0,1\}^n$ $P\{\nu_n = \alpha_n \,|\nu_{n-1} = \alpha_{n-1}, \ldots, \nu_1 = \alpha_1\} = P\{\nu_n = \alpha_n \,|\nu_{n-1} = \alpha_{n-1}, \ldots, \nu_{n-M} = \alpha_{n-M}\}$. Now, $\{(s_i, y_i)\}_{i=1}^\infty$ no longer forms a Markov chain, but $\left\{s_{r+lM}, y_{r+lM-M}^{r+lM}\right\}_{l=0}^\infty$, (for fixed $1 \leq r \leq M$) does form a non-homogeneous Markov chain. The following condition is the equivalent of $p > 0$ (when the BSC is concerned):

$$P\left\{\nu_{i-M+1}^i = \beta \,\middle|\nu_{i-2M+1}^{i-M} = \gamma\right\} \triangleq \rho_i(\beta,\gamma) \geq \rho \quad \forall \beta, \gamma \in \{0,1\}^M \tag{6}$$

This means, for instance, that the noise is not a periodic process.

---

[1] When $p = 0$ the current bit is given and need not be estimated. For the prediction problem with $p = 0$, an equivalent theorem has already been proved in [1]. $p = 0.5$ is a degenerate case, since there is no information in the observed data.

Theorem 3. *Let ν_1^n be a Markovian process that satisfies (6). Then for all $\theta \in \{0,1\}_1^\infty$ there exists a Markovian machine $m_{k,n} \in G(2^k)$ satisfying:*

$$e(\theta_1^n, m_{k,n}) \leq \min_{m \in \bar{M}(S)} e(\theta_1^n, m) + K'(\rho, S, M)\left(\alpha'(\rho, S, M)\right)^k + \frac{k}{n}. \tag{7}$$

where $|\alpha'(\rho, S, M)| < 1$ and $K'(\rho, S, M)$ are functions of ρ, S and M independent of θ_1^n.

Hence, $\mu(\theta) = \bar{e}(\theta)$.

3.2.3 Periodic FSMs

If $g^* = arg\min_{g \in G_S} e_g(\theta_1^n)$ is periodic with period T, it can be inferred that the time (modulo T) itself carries information about the empirical statistics of the sequence. In light of the proof of Theorem 1, we know for a fact that in the case of BSC, Markovian predictors asymptotically outperform FSMs, without an exclusion of periodic machines. It is conjectured that the performance of the optimal Markovian machine converges to $e_S(\theta_1^n)$ exponentially fast with k, also when the optimal FSM is periodic, and that the counter with which the extended Markovian machine is equipped is unnecessary.

Theorem 4. *Assume $p > 0$, then $\forall \theta \in \{0,1\}_1^\infty$ there exists an extended Markovian machine $m_{k,n} \in G(2^k, S!)$ satisfying :*

$$e(\theta_1^n, m_{k,n}) \leq \min_{m \in M(S)} e(\theta_1^n, m) + K''(p, S)\left(\alpha''(p, S)\right)^k + \frac{k}{n}. \tag{8}$$

where $|\alpha''(p, S)| < 1$ and $K''(p, S)$ are functions of p, S and independent of θ_1^n.

Hence, $\mu'(\theta) = e(\theta)$.

The proof is based on the fact that given the optimal finite state machine, one can construct a Markovian machine by feeding y_{t-k+1}^t into g with the initial state chosen to be an arbitrary state *belonging* to the subset of states that are visited at time instants i such that $i \mod T = (t-k) \mod T$.

4 A Sequential Scheme that Attains the S-State Filterability

In this section, we introduce an algorithm which asymptotically attains the performance of the non-sequential optimal state-fixed rules of a k-th order Markovian machine. This result is demonstrated in the framework of prediction, but it can easily be extended to estimation of the current bit.

If $\left\{EN_n^\theta(s,1), EN_n^\theta(s,0)\right\}_{s=1}^S$ were known in advance, then $e_g(\theta_1^n)$ could have been achieved by the prediction rule (4). If $\left\{N_n^\theta(s,1), N_n^\theta(s,0)\right\}_{s=1}^S$ were known in advance, even better performance could have been achieved with the same prediction rule applied to $\left\{N_n^\theta(s,1), N_n^\theta(s,0)\right\}_{s=1}^S$ instead of $\left\{EN_n^\theta(s,1), EN_n^\theta(s,0)\right\}_{s=1}^S$.

The main idea of the sequential scheme that is about to be introduced is that $N_n^\theta(s_t, x)$ is replaced by its current estimate given by $\hat{N}_{t-1}^\theta(s_t, x) = \frac{1-p}{1-2p} N_{t-1}^y(s_t, x) - \frac{p}{1-2p} N_{t-1}^y(s_t, |1-x|)$. In order to simplify the derivation, it is desirable to replace the estimator $\hat{N}_{t-1}^\theta(s_t, 1)$ which is a sum of dependent r.v.'s (due to the overlapping caused by the sliding window of the Markovian machine), with a sum of independent r.v.'s.

Let $J_t = \{i : i \mod (k+1) = t \mod (k+1)\}$ and let 1_A designate the indicator function, which equals 1 if the event A occurs and 0 otherwise. Define $M_t^y(s,x) = \sum_{i=1}^{t} 1_{i \in J_t} 1_{s_i = s, y_{i+1} = x}$ which is the joint count of previous occurrences of the state s_t in time indices in $\{t - l(k+1)\}_{l=1}^{\lfloor t/(k+1) \rfloor}$ followed by noisy bit x. Note that the r.v. $M_t^y(s,x)$ is a sum of *independent* r.v.'s since y_{i-k}^{i} is necessarily independent of y_{j-k}^{j} if $|j-i| > k$.

Consider the sequential prediction algorithm (4) applied to $M_{t-1}^y(s_t, x)$ instead of $EN_n^\theta(s_t, x)$ where $s_t = y_{t-k+1}^t$, and denote by $e_{k,seq}(\theta_1^n)$ the expected fraction of errors achieved by this algorithm.

Theorem 5. $\forall \theta_1^n$,

$$e_{k,seq}(\theta_1^n) - e_{g_k}(\theta_1^n) \le O\left(2^k k^{3/2} n^{-1/2}\right) \tag{9}$$

The derivation is partially based on [6], and it implements a lower bound on the Bayes envelope, $e_{g_k}(\theta_1^n)$ by the average loss associated with an impractical strategy that is allowed to access the next outcome y_{t+1} as well. The proof is omitted for the sake of brevity.

5 A Sequential Scheme that Attains the FS Filterability

In this section our aim is to introduce a sequential algorithm that asymptotically attains $e(\theta)$. In light of the theorems established so far, this aim is equivalent to sequentially achieving $\mu(\theta)$ or $\mu'(\theta)$ if prediction in the binary symmetric channel is concerned or sequential estimation/prediction in a more general channel are aimed at respectively.

In order to attain $\mu(\theta)$ sequentially, the order k of the Markovian machine must increase as more data is observed, otherwise if the order is bounded by some constant k^* the $(k^*+1) - th$ order filterability cannot be attained. We shall introduce two sequential algorithms that attain $\mu(\theta)$, but very similar schemes based on the same principal achieve $\mu'(\theta)$. We shall refer to the first algorithm as the "segmented scheme" and to the second as the incremental parsing (IP) scheme.

5.1 The Segmented Scheme

The first scheme is based on the very same principle of the equivalent scheme in [1]. Such a scheme can always be constructed provided that the performance of an offered sequential algorithm of order k converges to the desired (non-sequential) k-th order performance as time elapses, but is repeated here for the sake of completeness.

Recall that by Theorem 5 $e_{k,seq}(\theta_1^n) - e_{g_k}(\theta_1^n) \le O(2^k k^{3/2} n^{-1/2}) = \Delta(n,k)$. Divide y_1^n into k_n non-overlapping segments $\left\{y^{(k)}\right\}_{k=1}^{k_n}$ where the length of $y^{(k)}$ denoted n_k is a monotonically increasing sequence satisfying $\Delta(n_k, k) \triangleq \epsilon_k \to 0$. Note that, since $\epsilon_k \to 0$, the running average $\frac{1}{k_n}\sum_{k=1}^{k_n} \epsilon_k$ also converges to zero, i.e. $\lim_{k_n \to \infty} \frac{1}{k_n} \sum_{k=1}^{k_n} \epsilon_k = 0$. Apply the $k - th$ order sequential scheme described in Section 4 to $y^{(k)}$. Denote by $\theta^{(k)}$ and $\nu^{(k)}$ the clean and the noise segments that correspond to $y^{(k)}$. Since for all k, $e_{g_k}(\theta^{(k)})$ is attained (up to ϵ_k) by the sequential scheme applied to $y^{(k)}$, the expected fraction of errors attained by this scheme is equal to $\sum_{k=1}^{k_n} \frac{n_k}{n} e_{k,seq}(\theta^{(k)})$ and it can be shown that $\forall k' \le k_n$, $\sum_{k=1}^{k_n} \frac{n_k}{n} e_{k,seq}(\theta^{(k)}) \le e_{g_{k'}}(\theta_1^n) + \frac{k' n_{k'}}{2n} + \frac{1}{k_n} \sum_{k=1}^{k_n} \epsilon_k$ Since for fixed k', $\lim_{n \to \infty} \frac{k' n_{k'}}{2n} = 0$ and $\lim_{k_n \to \infty} \frac{1}{k_n} \sum_{k=1}^{k_n} \epsilon_k = 0$ this scheme attains $\mu(\theta)$.

5.2 Incremental Parsing Algorithm

The second scheme is based on the Incremental Parsing (IP) principle of the Lempel-Ziv '78 compression algorithm, in which the number of states increases as time elapses at an appropriate rate. The scheme is quite similar to the equivalent scheme in [1] and it can be viewed as a Markovian machine of increasing order. The incremental parsing of a sequence is a process of sequentially dividing a sequence into distinct phrases, such that each parsed phrase is the shortest phrase that has not been previously parsed. For instance: 0,00,1,001,10,11 is a parsed sequence. It is convenient to consider this procedure as a process of growing a tree (see [1]).

The process of incremental parsing can be viewed as a machine with increasing (with time) number of states, whose state at time instant t is the sequence of bits observed since the last phrase has terminated. For the sequence in the above example the state at time instant $t = 6$ is (00). The main difficulty in extending Theorem 5 to the IP procedure, is that the set of states that are visited by a corrupted sequence is random. This difficulty is overcome by introducing artificial randomization to the IP scheme.

Let $e_{seq}^{IP}(\theta_1^n)$ be the expected recurrence of errors achieved by the following scheme applied to a corrupted version of the individual sequence θ_1^n.

$$\hat{\theta}_{t+1} = \begin{cases} 0 & \tilde{N}_{t-1}^{\theta}(s_t, 0) > \tilde{N}_{t-1}^{\theta}(s_t, 1) \\ 1 & \tilde{N}_{t-1}^{\theta}(s_t, 1) > \tilde{N}_{t-1}^{\theta}(s_t, 0) \\ \mathit{arbitrary} & \tilde{N}_{t-1}^{\theta}(s_t, 1) = \tilde{N}_{t-1}^{\theta}(s_t, 0) \end{cases} \tag{10}$$

where s_t is the state at time instant t induced by the IP procedure, $\tilde{N}_t^{\theta}(s,x) = \hat{N}_t^{\theta}(s,x) + z_x(t)\,(N_t(s))^{1/2}$, $\hat{N}_t^{\theta}(s,x) = \frac{1-p}{1-2p} N_t^y(s,x) - \frac{p}{1-2p} N_t^y(s,|1-x|)$ is an estimator of $N_t^{\theta}(s,x)$, $\{z_0(t)\}_{t=1}^n$ and $\{z_1(t)\}_{t=1}^n$ are independent r.v.'s uniformly distributed over the interval $[0,1]$ and $N_t(s) = \sum_{i=1}^t 1_{s_i=s}$.

Theorem 6. *For all k and for all θ_1^n*

$$e_{seq}^{IP}(\theta_1^n) - e_{g_k}(\theta_1^n) \le O\left(\frac{1}{\sqrt{\log n}}\right) + O\left(\frac{k}{\log n}\right) \tag{11}$$

The proof is omitted for the sake of brevity.

6 Conclusions and Discussion

Theorems 1 - 4 state that the performance of any FSM can be approached arbitrarily close by a Markovian machine of sufficient order. The importance of this result is clear if one accepts the choice of FSMs as a good class of decision strategies, i.e., the finite number of states is accepted as a good candidate for a general limitation of the amount of resources of a practical algorithm. The difficulty that arises in competing with the unavailable optimal FSM is overcome by utilizing Markovian machines.

Theorem 5 is in compliance with previous results attained in the sequential compound decision problem, stating that in the finite alphabet finite strategy space, the difference between a sequential scheme and Bayes envelope decays like $O(n^{-1/2})$.

The two schemes that are introduced in Section 5, asymptotically attain the Markovian filterability of the infinite sequence and thus the FS filterability is attained as well. The segmentation applied in the segmented scheme, is somewhat artificial, since when a new segment starts, the whole memory is erased. The IP scheme gradually increases the Markovian order as a function of the observed data, and therefore is more plausible. It is argued that the randomization applied in the IP scheme is unnecessary, but it facilitates the derivation and therefore it is included in the scheme.

References

[1] M. Feder, N. Merhav, and M. Gutman, "Universal prediction of individual sequences," *IEEE Trans. Inform. Theory*, vol. 38, pp. 1258–1270, July 1992.

[2] H. Robbins, "Asymptotically subminimax solutions of compound statistical decision problems," in *Proc. 2nd Berkeley Symp. Math. Statist. Probab.*, pp. 131–148, 1951.

[3] J. F. Hannan and H. Robbins, "Asymptotic solutions of the compound decision problem for two completely specified distributions," *Ann. Math. Statist.*, vol. 26, pp. 37–51, 1955.

[4] J. F. Hannan, "Approximations to bayes risk in repeated plays," *in Contributions to the Theory of Games, Ann. Math. Studies (Princeton Univ., Princeton, NJ*, vol. 3, pp. 97–139, 1957.

[5] E. Samuel, "Asymptotic solutions of the sequential compound decision problem," *Ann. Math. Statist.*, vol. 34, pp. 1079–1094, 1963.

[6] J. V. Ryzin, "The sequential compound decision problem with $m \times n$ finite loss matrix," *Ann. Math. Statist.*, vol. 37, pp. 954–975, 1966.

[7] Y. Nogami, "The k-extended set-compound estimation problem in a nonregular family of distributions over $[\theta, \theta + 1)$," *Ann. Math. Statist.*, vol. 31A, pp. 169–176, 1979.

[8] S. B. Vardeman, "Admissible solutions of k-extended finite state set and sequence compound decision problems," *J. Multivariate Anal.*, vol. 10, pp. 426–441, 1980.

[9] J. Ziv and A. Lempel, "Compression of individual sequences via variable rate coding," *IEEE Trans. Inform. Theory*, vol. IT-24, pp. 530–536, September 1978.

[10] N. Merhav and M. Feder, "Universal schemes for sequential decision from individual data sequences," *IEEE Trans. Inform. Theory*, vol. 39, pp. 1280–1292, July 1993.

[11] N. Cesa-Bianchi, Y. Freund, D. Helmbold, D. Haussler, R. Schapire, and M. Warmuth, "How to use expert advice," *in Annu. ACM Symp. Theory of Computing*, pp. 382–391, 1993.

[12] A. Baruch and N. Merhav, "Universal filtering of individual sequences corrupted by noise," in *Proc. ISIT '98*, p. 331.

[13] A. Baruch, "Universal algorithms for sequential decision in the presence of noisy observations," Master's thesis, Technion -IIT, 1999.

[14] E. Seneta, *Non-negative Matrices and Markov Chains.* Springer-Verlag, second ed., 1981.

Fast RLS Laguerre Adaptive Filtering *

RICARDO MERCHED AND ALI H. SAYED

Electrical Engineering Department
University of California
Los Angeles, CA 90095

Abstract

The existing derivations of conventional fast RLS adaptive filters are intrinsically dependent on the shift structure in the input regression vectors. This structure arises when a tapped-delay line (FIR) filter is used as a modeling filter. In this paper, we show that a more general data structure is induced by other filter implementations, such as Laguerre-based filters and, more importantly, that an exact fast RLS algorithm can still be derived for such Laguerre-induced data structures. One of the benefits of working with a Laguerre basis is that fewer parameters can be used to model long impulse responses.

1 Introduction

Fast RLS adaptive filtering algorithms represent an attractive way to compute the least squares solution of growing length data efficiently. While the conventional RLS algorithm requires $\mathcal{O}(M^2)$ computations per sample, where M is the filter order, its fast counterparts require only $\mathcal{O}(M)$ operations. Examples of such fast schemes include the fast aposteriori error sequential technique (FAEST) [1], the fast transversal filter algorithm (FTF) [2], and least-squares lattice algorithms [3, 4, 5]. The latter class of algorithms deal with order-recursive structures, while the first two examples (FTF and FAEST) deal with fixed-order structures; both FTF and FAEST can also be viewed as special cases of a general fast estimation algorithm for state-space models, known as the (extended) Chandrasekhar recursions [6, 7].

The low complexity that is achieved by these algorithms is a direct consequence of the shift structure that is characteristic of regression vectors in FIR adaptive structures. This fact is evident in the conventional derivations of fast adaptive algorithms; all of which rely heavily on the shift structure. The arguments in [6, 7], however, have shown that fast RLS algorithms can still be derived for certain more general structures in the regression vectors, other than the shift structure. In this paper, we show that input regression vectors that arise from Laguerre-based networks satisfy the conditions required in [7] and, therefore, that fast RLS algorithms can still be derived in this context.

There are several important reasons to consider Laguerre basis functions instead of the usual FIR implementations. First, the use of Laguerre models (or more generally, orthonormal basis functions) to describe the dynamical behavior of a wide class of systems has been studied extensively in many recent works on system identification and control [8]-[11]. Second,

---

*This material was based on work supported in part by the National Science Foundation under Award CCR-9732376. The work of R. Merched was also in part supported by a fellowship from CAPES, Brazil.

the orthonormality property of Laguerre models offers many benefits in estimation problems, including better numerical conditioning of the data. Third, one of the primary motivations in using all-pass basis functions for adaptive filtering is the fact that it requires fewer parameters to model systems with long impulse responses.

In echo cancelation applications, for example, a long FIR filter may be necessary to model the echo path and adaptive IIR techniques have been proposed as possible alternatives (e.g., [12]-[13]). These techniques are nevertheless known to face stability problems due to the arbitrary pole locations during filter operation. Laguerre-domain adaptive filtering can offer an attractive alternative since, in this case, the pole location is fixed. It has already been suggested for echo cancelation and equalization applications in the works [14, 15]. However, these earlier contributions rely only on a slow RLS-type form for Laguerre adaptive filtering that requires $\mathcal{O}(M^2)$ operations. Fast $\mathcal{O}(M)$ Laguerre adaptive filters have been proposed before in the literature, but only as extensions of the classical gradient adaptive lattice (GAL) algorithm, which relies on LMS-type recursions and not on RLS-type recursions (see [16]).

In this paper, we show that a fast $\mathcal{O}(M)$ Laguerre adaptive filter can actually be derived in the least-squares domain, thus leading to a fast RLS Laguerre adaptive scheme. One of the advantages of insisting on a least-squares-based adaptive scheme is that these tend to exhibit faster convergence rates and smaller misadjustments than gradient-based adaptive schemes. The algorithm of this paper will be presented in a compact array form, where arrays of numbers are transformed via suitably chosen elementary rotations. Such array forms exhibit better numerical properties than conventional descriptions in terms of explicit sets of equations.

2 The RLS Algorithm

Given a column vector $y \in \mathbb{C}^{N+1}$ and a data matrix $H \in \mathbb{C}^{(N+1)\times M}$, the exponentially-weighted least squares problem seeks the column vector $w \in \mathbb{C}^M$ that solves

$$\min_{w} \left[\lambda^{N+1} w^* \Pi_0^{-1} w + (y - Hw)^* W^{-1} (y - Hw)\right] . \tag{1}$$

The matrix Π_0 is a positive-definite regularization matrix, and $W = (\lambda^N \oplus \lambda^{N-1} \oplus \cdots \oplus 1)$ is a weighting matrix that is defined in terms of a forgetting factor λ. In tracking problems, typically we have $0 \ll \lambda < 1$. The symbol $*$ denotes complex conjugate transposition.

The individual entries of y will be denoted by $\{d(i)\}$, and the individual rows of H will be denoted by $\{u_i\}$,

$$y = \begin{bmatrix} d(0) \\ d(1) \\ \vdots \\ d(N) \end{bmatrix} , \quad H = \begin{bmatrix} u_0 \\ u_1 \\ \vdots \\ u_N \end{bmatrix} .$$

The RLS algorithm computes the optimal solution of problem (1) recursively as follows:

$$\begin{aligned} w_{i+1} &= w_i + g_{i+1}[d(i+1) - u_{i+1} w_i] , \quad w_{-1} = 0 , && (2) \\ g_{i+1} &= \lambda^{-1} P_i u_{i+1}^* \gamma(i+1) , && (3) \\ \gamma^{-1}(i+1) &= 1 + \lambda^{-1} u_{i+1} P_i u_{i+1}^* , && (4) \\ P_{i+1} &= \lambda^{-1} P_i - g_{i+1} \gamma^{-1}(i+1) g_{i+1}^* , \quad P_{-1} = \Pi_0 . && (5) \end{aligned}$$

The computation of the gain vector g_{i+1} in the above solution relies on the propagation of the Riccati variable P_i. This method of computation requires $\mathcal{O}(M^2)$ operations per iteration.

Fast $\mathcal{O}(M)$ RLS schemes, on the other hand, avoid the propagation of P_i and therefore evaluate the necessary gain vector in an alternative more efficient manner. It turns out that the choice of Π_0 plays a crucial role in the derivation of such fast schemes, as we now explain.

Assume for the time being that there exists a square matrix Φ that relates two successive regression vectors as

$$u_i = u_{i+1}\Phi \; . \tag{6}$$

Then using Eq. (2) we can relate the two successive gain vectors $\{g_i, g_{i+1}\}$ as follows. Note that

$$\Phi g_i \gamma^{-1}(i) = \lambda^{-1}\Phi P_{i-1}u_i^* \; = \; \lambda^{-1}\Phi P_{i-1}\Phi^* u_{i+1}^* \; .$$

Subtracting this equality from $g_{i+1}\gamma^{-1}(i+1) = \lambda^{-1}P_i u_{i+1}^*$, we find that

$$g_{i+1}\gamma^{-1}(i+1) = \Phi g_i \gamma^{-1}(i) \; + \; \lambda^{-1}(P_i - \Phi P_{i-1}\Phi^*)u_{i+1}^* \; . \tag{7}$$

This equation shows that in order to update the (scaled) gain vector from time i to time $i+1$, it is not necessary to evaluate the individual $\{P_i's\}$ but rather the differences

$$\nabla_{\{P_i,\Phi\}} = P_i - \Phi P_{i-1}\Phi^* . \tag{8}$$

It turns out that such differences can be updated fast for certain choices of Π_0 and Φ (as explained in [6, 7] in a more general state-space context), and this fact can be used to derive fast RLS schemes for input data vectors with or without shift structure.

3 Fast RLS Laguerre Algorithm

Thus consider the Laguerre-based model of Figure 1 where

$$H_0(z) = \frac{\sqrt{1-a^2}}{1-az^{-1}} \quad \text{and} \quad H(z) = \frac{z^{-1}-a}{1-az^{-1}} \; , \quad |a| < 1. \tag{9}$$

Note that $H(z)$ is a first-order all-pass system and that, unlike a general IIR structure, the poles of the Laguerre-based model are fixed at a.[1] [The choice of a can be optimized off-line according to some criterion.] The input to the Laguerre filter at time i is denoted by $x(i)$, and the coefficients that combine the outputs of the successive sections $\{H_0(z), H(z)\}$ are denoted by $\{w_k\}$.

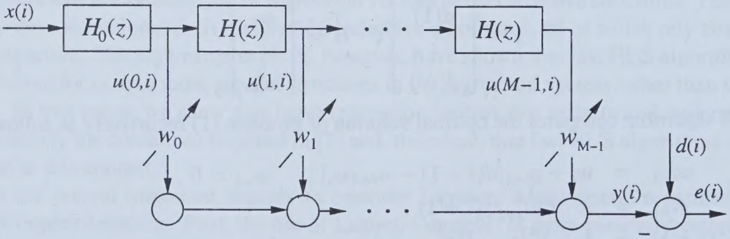


Figure 1: *A transversal Laguerre structure for adaptive filtering.*

---

[1] The case of complex poles can also be handled. For simplicity, we assume real a.

Let

$$u_i \triangleq [u(0,i) \quad u(1,i) \quad \cdots \quad u(M-1,i)] \,,$$

denote the regression vector at time i. Observe that the individual entries of u_i are not shifted versions of each other; instead, they are the outputs of the successive sections $\{H_0(z), H(z)\}$ at time i. Let us now verify that two successive regression vectors $\{u_i, u_{i+1}\}$ lead to a relation similar to (but not exactly of the same form as) Eq. (6) for some Φ. [Later we shall show that we can handle the slight discrepancy in the relation by properly defining extended vectors.]

Using (9) we can write difference equations relating the entries of two successive regression vectors u_i and u_{i+1}. For example, the first three entries satisfy the relations:

$$\begin{aligned}
u(0,i) &= \frac{1}{a}u(0,i+1) - \frac{\sqrt{1-a^2}}{a}x(i+1) \\
u(1,i) &= \frac{1}{a}u(1,i+1) + \left(1-\frac{1}{a^2}\right)u(0,i+1) + \frac{\sqrt{1-a^2}}{a^2}x(i+1) \\
u(2,i) &= \frac{1}{a}u(2,i+1) + \left(1-\frac{1}{a^2}\right)u(1,i+1) - \frac{1}{a}\left(1-\frac{1}{a^2}\right)u(0,i+1) - \frac{\sqrt{1-a^2}}{a^3}x(i+1)
\end{aligned}$$

In matrix form, we can express the above difference equations as (say, for $M=4$),

$$u_i = [u_{i+1} \quad x(i+1)] \begin{bmatrix} T \\ v \end{bmatrix} , \tag{10}$$

where T is the $(M \times M)$ upper triangular Toeplitz matrix

$$T = \begin{bmatrix} \frac{1}{a} & \left(1-\frac{1}{a^2}\right) & -\frac{1}{a}\left(1-\frac{1}{a^2}\right) & \frac{1}{a^2}\left(1-\frac{1}{a^2}\right) \\ 0 & \frac{1}{a} & \left(1-\frac{1}{a^2}\right) & -\frac{1}{a}\left(1-\frac{1}{a^2}\right) \\ 0 & 0 & \frac{1}{a} & \left(1-\frac{1}{a^2}\right) \\ 0 & 0 & 0 & \frac{1}{a} \end{bmatrix} , \tag{11}$$

and v is the $(M \times 1)$ row vector

$$v = \begin{bmatrix} -\frac{\sqrt{1-a^2}}{a} & \frac{\sqrt{1-a^2}}{a^2} & -\frac{\sqrt{1-a^2}}{a^3} & \frac{\sqrt{1-a^2}}{a^4} \end{bmatrix} . \tag{12}$$

Except for the additional term $x(i+1)$, relation (10) is of the same form as (6). This slight difference in the nature of the relations can be handled easily by properly defining extended quantities.

Thus note that using Eq. (2), we can write

$$\begin{bmatrix} g_{i+1}\gamma^{-1}(i+1) \\ 0 \end{bmatrix} = \lambda^{-1} \begin{bmatrix} P_i & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} u_{i+1}^* \\ x^*(i+1) \end{bmatrix} . \tag{13}$$

Likewise, we have

$$\begin{bmatrix} T \\ v \end{bmatrix} g_i\gamma^{-1}(i) = \lambda^{-1} \begin{bmatrix} T \\ v \end{bmatrix} P_{i-1}u_i^* \; = \; \lambda^{-1} \begin{bmatrix} T \\ v \end{bmatrix} P_{i-1} \, [T^* \quad v^*] \begin{bmatrix} u_{i+1}^* \\ x^*(i+1) \end{bmatrix} . \tag{14}$$

Subtracting (13) from (14), we obtain

$$\begin{bmatrix} g_{i+1}\gamma^{-1}(i+1) \\ 0 \end{bmatrix} - \begin{bmatrix} T \\ v \end{bmatrix} g_i\gamma^{-1}(i) = \lambda^{-1} \left(\begin{bmatrix} P_i & 0 \\ 0 & 0 \end{bmatrix} - \begin{bmatrix} T \\ v \end{bmatrix} P_{i-1} \, [T^* \quad v^*] \right) \begin{bmatrix} u_{i+1}^* \\ x^*(i+1) \end{bmatrix} \tag{15}$$

This shows that in order to update the (scaled) gain vector from g_i to g_{i+1}, we only need to know how to update the difference

$$\nabla_{\{P_i,\Phi\}} \triangleq \begin{bmatrix} P_i & 0 \\ 0 & 0 \end{bmatrix} - \Phi \begin{bmatrix} P_{i-1} & 0 \\ 0 & 0 \end{bmatrix} \Phi^* ,$$

where we are now defining the $(M+1) \times (M+1)$ square matrix Φ as

$$\Phi = \begin{bmatrix} T & 0 \\ v & 0 \end{bmatrix} .$$

We shall now verify that it is possible to update the differences $\nabla_{\{P_i,\Phi\}}$ efficiently. For this purpose, we first note that for the case of pre-windowed input data (i.e., $x(i) = 0$ for $i \leq 0$ and zero initial conditions), we obtain at the initial time instant $i = 0$,

$$\nabla_{\{P_0,\Phi\}} = \begin{bmatrix} \lambda^{-1}\Pi_0 & 0 \\ 0 & 0 \end{bmatrix} - \Phi \begin{bmatrix} \Pi_0 & 0 \\ 0 & 0 \end{bmatrix} \Phi^* . \tag{16}$$

Assume that Π_0 is chosen such that the above difference has low rank (say r, where r is independent of M — see Sec. 5). We can then factor $\nabla_{\{P_0,\Phi\}}$ as

$$\nabla_{\{P_0,\Phi\}} = L_0 J L_0^* , \tag{17}$$

where L_0 is $(M+1) \times r$ and J is a signature matrix with as many $\pm 1's$ as $\nabla_{\{P_0,\Phi\}}$ has positive or negative eigenvalues. It will then follow that the rank of all successive differences, $\nabla_{\{P_i,\Phi\}}$ for $i > 0$, will not exceed r and, more importantly, that the inertia of all these successive differences can also be taken as J. In other words, by forcing the initial difference to have low rank, and a certain inertia, we end up forcing all successive differences to have a similar property. This fact is essential to the derivation of a fast algorithm.

To establish the above claims we proceed by induction. Thus assume that the difference $\nabla_{\{P_i,\Phi\}}$ at time i can be factored as $\nabla_{\{P_i,\Phi\}} = L_i J L_i^*$ for some $(M+1) \times r$ matrix L_i. Define further, for compactness of notation, the extended quantities:

$$\bar{u}_{i+1} \triangleq [u_{i+1} \quad x(i+1)] , \quad \bar{g}_i \triangleq \begin{bmatrix} g_i \\ 0 \end{bmatrix} , \quad \bar{k}_i = \bar{g}_i \gamma_i^{-1/2}(i) , \quad \text{and} \quad \bar{P}_i \triangleq \begin{bmatrix} P_i & 0 \\ 0 & 0 \end{bmatrix} .$$

Now implement a $(1 \oplus J)$-unitary transformation matrix Θ_i (i.e., Θ_i satisfies $\Theta_i (1 \oplus J) \Theta_i^* = (1 \oplus J)$) that transforms the following pre-array to the form

$$\begin{bmatrix} \gamma^{-1/2}(i) & \frac{1}{\sqrt{\lambda}} \begin{bmatrix} u_{i+1} & x(i+1) \end{bmatrix} L_i \\ \Phi \bar{k}_i & \frac{1}{\sqrt{\lambda}} L_i \end{bmatrix} \Theta_i = \begin{bmatrix} s & 0 \\ m & C \end{bmatrix} , \tag{18}$$

where s is a scalar, m is a column vector, and C is a matrix. Then we claim that we can make the identifications

$$s = \gamma^{-1/2}(i+1), \quad m = \bar{k}_{i+1} , \quad C = L_{i+1} ,$$

and also show that $\nabla_{\{P_{i+1},\Phi\}} = CJC^*$. [This last equation means that the inertia of $\nabla_{\{P_{i+1},\Phi\}}$ can also be taken as J, and that the above array algorithm provides the desired low rank factor L_{i+1} as well.]

To determine the $\{s, m, C\}$ as above, all we need to do is simply compare entries on both sides of the following equality (which holds in view of the $(1 \oplus J)$-unitarity of Θ_i):

$$\begin{bmatrix} \gamma^{-1/2}(i) & \frac{1}{\sqrt{\lambda}}\begin{bmatrix} u_{i+1} & x(i+1) \end{bmatrix} L_i \\ \Phi\bar{k}_i & \frac{1}{\sqrt{\lambda}}L_i \end{bmatrix} \begin{bmatrix} 1 & \\ & J \end{bmatrix} \begin{bmatrix} \gamma^{-1/2}(i) & \frac{1}{\sqrt{\lambda}}\begin{bmatrix} u_{i+1} & x(i+1) \end{bmatrix} L_i \\ \Phi\bar{k}_i & \frac{1}{\sqrt{\lambda}}L_i \end{bmatrix}^* =$$

$$\begin{bmatrix} s & 0 \\ m & C \end{bmatrix} \begin{bmatrix} 1 & \\ & J \end{bmatrix} \begin{bmatrix} s & 0 \\ m & C \end{bmatrix}^* . \tag{19}$$

The resulting fast algorithm can be summarized as follows.

Algorithm 1 (Fast Array Laguerre Algorithm) *Consider input regression vectors u_i arising from the Laguerre structure of Figure 1. The solution to the minimization problem (1) can be recursively computed as follows. Start with $w_{-1} = 0$, $\gamma^{-1/2}(0) = 1$, $g_0 = 0$, L_0 and J from the factorization (17), and repeat for each $i > 0$,*

$$\left[\begin{bmatrix} T \\ v \end{bmatrix} \begin{matrix} \gamma^{-1/2}(i) \\ g_i \gamma_i^{-1/2}(i) \end{matrix} \quad \begin{matrix} \frac{1}{\sqrt{\lambda}}[u_{i+1} \;\; x(i+1)] L_i \\ \frac{1}{\sqrt{\lambda}} L_i \end{matrix} \right] \Theta_i = \left[\begin{matrix} \gamma^{-1/2}(i+1) & 0 \\ \begin{bmatrix} g_{i+1}\gamma^{-1/2}(i+1) \\ 0 \end{bmatrix} & L_{i+1} \end{matrix} \right],$$

where Θ_i is a $(1 \oplus J)$-unitary matrix that produces the zero entries in the post-array, and L_i is $(M+1) \times r$. The transformation matrix Θ_i can be implemented according to the procedure described below. The upper triangular Toeplitz matrix T and the row vector v are as defined in Eqs. (11) and (12). Moreover,

$$w_{i+1} = w_i + g_{i+1}[d(i+1) - u_{i+1}w_i].$$

◇

4 Implementation Issues

Although from a theoretical point of view, any $(1 \oplus J)$-unitary matrix Θ_i that produces the zero entries in the first row of the post-array will do, we have noticed that different implementations lead to different numerical behavior. To see this, consider for simplicity the case $M = 3$ and $J = (1 \oplus -1)$. Then the pre- and post-arrays will be of the generic forms:

$$\begin{bmatrix} \times & \times & \times \\ \times & \times & \times \\ \times & \times & \times \\ \times & \times & \times \\ \times & \times & \times \end{bmatrix} \Theta = \begin{bmatrix} \times & 0 & 0 \\ \times & \times & \times \\ \times & \times & \times \\ \times & \times & \times \\ 0 & \times & \times \end{bmatrix} .$$

In order to create the zero pattern in the first row of the post array, the $(1 \oplus J)$-unitary rotation Θ can be evaluated based only on the first row of the pre-array, which means that only the information needed to update $\gamma^{-1}(i)$ to $\gamma^{-1}(i+1)$ is required in order to generate the matrix Θ. In other words, in this case no information from the other equations is used to update the rest of the entries of the array. We have observed in simulations (see Sec. 6) that even for $\lambda = 1$, this type of transformation can cause the algorithm to diverge in finite precision.

To mitigate this problem, we propose to apply Θ as follows. First create one zero in the first row of the post-array by means of a circular rotation, using the entries $(0, 0)$ and $(0, 1)$ (as indicated by the arrows) in the pre-array:

$$\begin{array}{c} \begin{array}{ccc} \downarrow & \downarrow & \end{array} \\ \begin{bmatrix} \otimes & \otimes & \times \\ \times & \times & \times \\ \times & \times & \times \\ \times & \times & \times \\ \times & \times & \times \end{bmatrix} \end{array} \xrightarrow{\text{Givens}} \begin{bmatrix} \times & 0 & \times \\ \times & \times & \times \\ \times & \times & \times \\ \times & \times & \times \\ \times & \times & \times \end{bmatrix} .$$

Now, note that the additional hyperbolic rotation that is needed to zero out the remaining entry in the first row of the post-array, also needs to create a zero in the entry $(M+1, 0)$ of the post-array. Rather than determining this hyperbolic rotation by using the entries $(0, 0)$ and $(0, 3)$ of the *first* row of the pre-array, we propose instead to determine the hyperbolic rotation by using the entries $(M+1, 0)$ and $(M+1, 3)$ of the *last* row of the pre-array. That is,

$$\begin{array}{c} \begin{bmatrix} \times & \times & \times \\ \times & \times & \times \\ \times & \times & \times \\ \times & \times & \times \\ \otimes & \times & \otimes \end{bmatrix} \\ \begin{array}{ccc} \uparrow & & \uparrow \end{array} \end{array} \xrightarrow{\text{Hyperbolic}} \begin{bmatrix} \times & 0 & 0 \\ \times & \times & \times \\ \times & \times & \times \\ \times & \times & \times \\ 0 & \times & \times \end{bmatrix} .$$

This choice is actually more reasonable, since in this case, the rotation matrix Θ is determined by using all the equations that constitute the fast recursions. We have verified by simulations that this method of constructing Θ presents no signs of instability for the case $\lambda = 1$.

On another matter, the algorithm also requires the evaluation of a matrix-vector product of the form Tp for some vector p. Now since the matrix T is upper triangular Toeplitz, this product amounts to a convolution operation. Actually, the entries of Tp can be obtained by filtering the (reversed) entries of p through the maximum-phase all-pass filter $Q(z) = (z^{-1} + a^{-1})/(1 + a^{-1}z^{-1})$.

Another possibility for the evaluation of Tp is to rely on fast transform techniques. One well-known technique (known as the *extension approach*) is to embed the Toeplitz matrix into a larger $n \times n$ circulant matrix, which can then be diagonalized by the DFT matrix. A second technique, known as the *decomposition approach*, is to express T as the sum of circulant and *skew*-circulant matrices. Both techniques can also be extended to trigonometric and Hartley transforms (see [17]), which are better suited for cases with real-valued data.

5 Choice of the Regularization Matrix

We now examine choices for the positive-definite matrix Π_0 such that the rank of the initial difference $\nabla_{\{P_0,\Phi\}}$ is low (2 or 3). For any such choice of Π_0, all successive differences $\nabla_{\{P_i,\Phi\}}$ will also have low rank with the same inertia. [The actual value of Π_0 is not needed in the fast algorithm, but rather the corresponding factors $\{L_0, J\}$. This initial calculation is performed off-line and can be assigned to overhead costs.]

Assume first that $\lambda = 1$. Then the choice $\Pi_0 = \delta I$, for any $\delta > 0$, leads to a difference $\nabla_{\{P_0,\Phi\}}$ of the form

$$\nabla_{\{P_0,\Phi\}} = \delta\left(\begin{bmatrix} \Pi_0 & 0 \\ 0 & 0 \end{bmatrix} - \begin{bmatrix} T & 0 \\ v & 0 \end{bmatrix} \begin{bmatrix} \Pi_0 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} T^* & v^* \\ 0 & 0 \end{bmatrix} \right) = \delta \begin{bmatrix} I - TT^* & -Tv^* \\ -vT^* & -\|v\|^2 \end{bmatrix} .$$

This difference can be easily seen to have rank 2. This is because $I - TT^*$ has rank one, viz., $I - TT^* = -\alpha^2 cc^*$ with

$$\alpha = \frac{\sqrt{1-a^2}}{a^M} , \qquad c = \text{col}\{1, -a, a^2, -a^3, \ldots, (-a)^{M-1}\} ,$$

and the vector Tv^* is collinear with c, since $Tv^* = -\beta c$ with $\beta = \alpha/a^M$. Therefore, $\nabla_{\{P_0,\Phi\}}$ has rank 2, with

$$J = \begin{bmatrix} 1 & \\ & -1 \end{bmatrix}, \quad L_0 = \sqrt{\delta} \cdot \begin{bmatrix} 0 & \alpha c \\ 1 & -\frac{1}{a^M} \end{bmatrix}.$$

Let us now consider the case $\lambda \neq 1$, for which the difference $\nabla_{\{P_0,\Phi\}}$ becomes

$$\nabla_{\{P_0,\Phi\}} = \begin{bmatrix} \lambda^{-1}\Pi_0 - T\Pi_0 T^* & -T\Pi_0 v^* \\ -v\Pi_0 T^* & -v\Pi_0 v^* \end{bmatrix}. \tag{20}$$

We want to argue that there are choices of Π_0 that lead to at least a rank 3 difference. This can be seen as follows. Assume first that $\sqrt{\lambda}/a < 1$. Then any vector b such that the pair $(\lambda^{1/2}T, b)$ is controllable will result in a positive-definite solution Π_0 for the Lyapunov equation

$$\lambda^{-1}\Pi_0 - T\Pi_0 T^* = bb^*. \tag{21}$$

If, on the other hand, $\sqrt{\lambda}/a > 1$, then any vector b such that the pair $(\lambda^{1/2}T, b)$ is controllable, will result in a positive-definite solution Π_0 for the following Lyapunov equation

$$\lambda^{-1}\Pi_0 - T\Pi_0 T^* = -bb^*. \tag{22}$$

In either case, the point is that we can choose a b (and consequently, Π_0) such that the difference $\lambda^{-1}\Pi_0 - T\Pi_0 T^*$ has rank one (and inertia 1 or -1 depending on whether $\sqrt{\lambda}/a$ is smaller or larger than one). It then follows that the rank of $\nabla_{\{P_0,\Phi\}}$ in (20) will be 3 and its inertia will be $\{\mu, -1, 1\}$, where $\mu = +1$ or $\mu = -1$. [We may remark that once a vector b has been chosen, it can be scaled arbitrarily by a number $\sqrt{\delta}$. This helps guarantee that the resulting Π_0 will be sufficiently large, as is often required in such least-squares applications.]

[Also, if one insists on a rank 2 matrix $\nabla_{\{P_0,\Phi\}}$, then we need to find a b such that $-T\Pi_0 v^*$ is collinear with b as well.]

6 Simulation Results

The simulation results in this section were obtained by averaging over 10 runs in a system identification scenario with 50 dB signal-to-noise ratio. The input to the unknown model was taken as colored noise.

Figure 2 shows the results of two experiments with $\lambda = 1$ and a Laguerre filter with 5 taps and poles at $a = 0.5$. The regularization matrix was chosen as $\Pi_0 = 10^5 I$. The figure shows two learning curves corresponding to the two different implementations of the rotation matrices Θ_i, as explained in Sec. 4. We can see from the figure that the classical method of implementation of the rotations leads to divergence in a few iterations. Figure 3 compares the performance of the fast Laguerre filter with the fast FIR filter (both implemented in array forms). The model used for the system identification example in Figure 3 is the same from [16]):

$$G(z) = 0.0017\frac{z^{-1}(1 + 0.673z^{-1})}{(1 - 0.368z^{-1})(1 - 0.819z^{-1})(1 - 0.995z^{-1})}.$$

The Laguerre filter was implemented with 6 taps and the FIR filter with 500 taps. Both filters used $\lambda = 0.999$. We see from the figure that the Laguerre-RLS algorithm presents faster convergence and achieves a lower MSE level compared to the fast FIR-RLS algorithm.

We should also mention that for $\lambda < 1$, the fast recursions may encounter convergence difficulties. For $\lambda = 1$, and by using the proposed method for implementing the rotations,

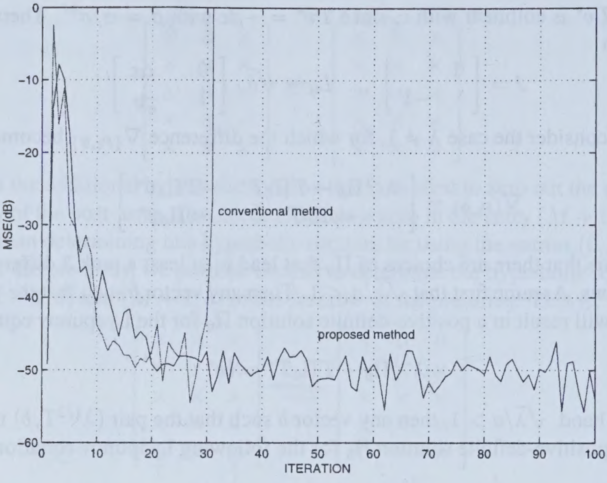


Figure 2: *Learning curves for the fast* RLS *Laguerre using two forms of hyperbolic transformations.*

we did not notice such problems even over long simulations. [Actually, even the fast array algorithm for regression vectors with shift structure is also unstable for $\lambda = 1$ if the rotations are not implemented with care.] Methods to stabilize the recursions in the general case are under investigation.

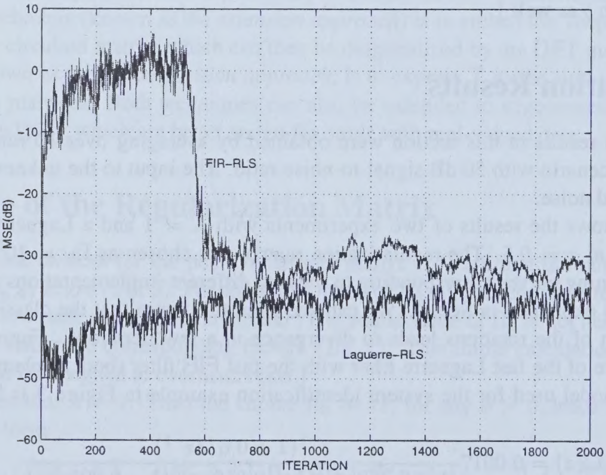


Figure 3: *Comparison of a 6-tap fast Laguerre filter with a 500-tap fast* FIR *filter.*

References

[1] G. Carayannis, D. Manolakis, and N. Kalouptsidis, "A fast sequential algorithm for least squares filtering and prediction," *IEEE Trans. on Acoust., Speech, Signal Proc.*, vol. ASSP-31, pp. 1394–1402, December 1983.

[2] J. Cioffi and T. Kailath, "Fast recursive-least-squares transversal filters for adaptive filtering," *IEEE Trans. on Acoust., Speech Signal Processing*, vol. ASSP-32, pp. 304-337, April 1984.

[3] D. T. L. Lee, M. Morf, and B. Friedlander, "Recursive least-squares ladder estimation algorithms," *IEEE Trans. on Circuits. and Syst.*, no. 6, pp. 467–481, June 1981.

[4] B. Friedlander, "Lattice filters for adaptive processing," *Proc. IEEE*, vol. 70, pp. 829–867, Aug. 1982.

[5] H. Lev-Ari, T. Kailath, and J. Cioffi, "Least-squares adaptive lattice and transversal filters: A unified geometrical theory," *IEEE Trans. on Inform. Theory*, vol. IT-30, pp. 222-236, March 1984.

[6] A. H. Sayed and T. Kailath, "Extended Chandrasekhar recursions," *IEEE Trans. on Automatic Control*, vol. AC-39, no. 3, pp. 619-623, March 1994.

[7] A. H. Sayed and T. Kailath, "A state-space approach to adaptive RLS filtering," *IEEE Signal Processing Magazine*, vol. 11, no. 3, pp. 18-60, Jul. 1994.

[8] B. Wahlberg, "System Identification using Laguerre models," *IEEE Trans, Automat. Control.*, vol. 36, pp. 551-562, May. 1991.

[9] P. M. Makila, "Approximation of stable systems by Laguerre filters,"*Automatica*, vol. 26, pp. 333-345, Feb. 1990.

[10] G. A. Dumont and C. C. Zervos, "Adaptive control based on orthonormal series representation," *Proceedings of* 2^{nd} *IFAC Workshop on Adaptive Systems in Control and Signal Processing*, Lund, Sweden, pp.371-376, 1988.

[11] P. Heuberger, B. Ninness, T. Oliveira e Silva, P. Van den Hof and B. Wahlberg, "Modeling and Identification with Orthogonal Basis Functions," 36^{th} IEEE CDC Pre-Conference Workshop no. 7, San Diego, CA, Dec. 1997.

[12] J. J. Shynk, "Adaptive IIR filtering," *IEEE AASSP Magazine*, vol. 6, pp. 4-21, April 1989.

[13] P. A. Regalia, *Adaptive IIR Filtering in Signal Processing and Control*, Marcel Dekker, NY, 1995.

[14] J. W. Davidson and D. D. Falconer, "Reduced complexity echo cancelation using orthonormal functions," *IEEE Trans. on Circuits Syst.*, vol. 38, no. 1, pp. 20-28, Jan. 1991.

[15] L. Salama and J. E. Cousseau, "Efficient echo cancelation based on an orthogonal adaptive IIR realization," *ITS'98 Proceedings. SBT/IEEE Int. Telecom. Symposium*, São Paulo, Brazil, Aug. 1998.

[16] Z. Fejzo and H. Lev-Ari, "Adaptive Laguerre-lattice filters," *IEEE Trans. on Signal Processing*, vol. 45, no. 12, pp. 3006-3016, Dec. 1997.

[17] G. Heinig and K. Rost, "Representations of Toeplitz-plus-Hankel matrices using trigonometric transformations with application to fast matrix-vector multiplication," *Linear Algebra and its Applications*, vol. 257-276, pp. 225-248, 1998.

Machine Learning Applications in Grid Computing

George Cybenko, Guofei Jiang and Daniel Bilar

Thayer School of Engineering
Dartmouth College
Hanover, NH 03755, USA
gvc@dartmouth.edu, guofei.jiang@dartmouth.edu

Abstract

The development of the World Wide Web has changed the way that we think about information. Information on the web is distributed, updates are made asynchronously and resources come and go online without centralized control. Global networking will similarly change the way we think about and perform computation. *Grid computing* refers to computing in a distributed networked environment in which computing and data resources are located throughout a network. Grid computing is enabled by an *infrastructure* that allows users to locate computing resources and data products dynamically during a computation. In order to locate resources dynamically in a grid computation, a grid application program consults a broker or matchmaker agent that uses keywords and ontologies to specify grid services. However, we believe that keywords and ontologies cannot be defined or interpreted precisely enough to make brokering and matchmaking between agents sufficiently robust in a truly distributed, heterogeneous computing environment. To this end, we introduce the concept of functional validation. Functional validation goes beyond the symbolic negotiation level of brokering and matchmaking, to the level of validating actual functional performance of grid services. In this paper, we present the functional validation problem in grid computing and apply basic machine learning theory such as PAC learning and Chernoff bounds to solve the sample size problem that arises. Furthermore, in order to reduce network traffic and speedup the validation process, we describe the use of Dartmouth D'Agents technology to implement a general mobile functional validation agent system which can be integrated into a grid computing infrastructures as a standard grid service.

1. Introduction

There are currently several efforts are underway to build *computational grids*, such as Globus [1], Infospheres [2] and the DARPA CoABS [3] . These projects are developing the fundamental technology that is needed to build computational grids, execution environments that enable an application to integrate geographically distributed instruments, sensors, data products, displays, and computational and information resources. Such grid computations may link tens or hundreds of these resources. The vision is that these grid infrastructures will connect multiple regional and national computational grids, creating a universal source of pervasive and dependable computing power that supports dramatically new classes of applications.

Acknowledgements: This work was partially supported by Air Force Office of Scientific Research grants F49620-97-1-0382, National Science Foundation grant CCR-9813744 and DARPA contract F30602-98-2-0107.

A fundamental capability required in such a grid is a directory service or broker that dynamically matches user requirements with available resource. On the web, this capability is provided by search engines that index web pages and implement retrieval services. Whereas humans are typically the consumers of web information, grid agents will be the producers and consumers of grid resources with humans occasionally steering or interpreting the computations.

A grid computation that needs a computational service, such as for example the solution to structured linear system or a multidimensional Fourier transform, will locate the required service by consulting a distributed object request broker or a matchmaker service. For example, CORBA is an infrastructure for implementing distributed applications and provides a broker as a key component [4]. Jini[5] is designed for deploying and using services in a network and enables the construction of dynamic, flexible, and robust systems from independent distributed components and it also can provide a nice lookup service. An object request broker (ORB) not only locates a component or object that performs the required service but also mediates communication between the client and the service. In standard terminology, a matchmaker is an ORB with reduced capability. A matchmaker merely locates remote services but does not mediate communications between client and server agents. In the matchmaker framework, a client and the remote service that it invokes communicate directly once their locations are made known by the matchmaker service.

2. Grid Computing Services and Functional Validation

A schematic of part of a grid computation is shown in Figure 1. Remote service providers will publish their services' catalog on ORB or matchmaker agents. When clients need some computing services, they will make a request to an ORB or matchmaker agent. The request is made based on some keywords and possibly parameter lists for invoking the remote computing resources. The ORB or matchmaker consults its service catalog and returns with several remote service candidates and specifies their returning value structures. Then clients can negotiate with these service candidates and ask for the appropriate computing service. Just like web search engines, ORB and matchmaker agents will use keywords and ontologies to specify remote computing services. Ontologies specify a domain and keywords specify functionality within that domain. For example, ontologies are envisioned for signal processing, ocean modeling, image processing, weather modeling and so on. Within an ontology, keywords such as "Fourier transform" and "linear system solver" will have possibly domain specific meanings. Several systems have been proposed for implementing such ontological matching [6][7].

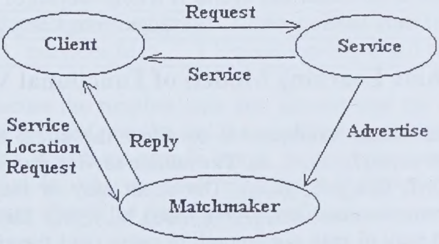


Figure 1: The prototype of grid computing services

Note however, there are literally dozens of different algorithms for implementing Discrete Fourier Transforms [8]. Different algorithms make different assumptions about the symmetries of the input vector and order the output in a variety of ways. Some algorithms may be only able to transform the input vector of certain dimensions. The actual numerical computations carried out vary from algorithm to algorithm so that different round-off errors are accumulated leading to slightly different answers. Moreover, different numerical implementations of some basic computations in an can lead to different computational speed, different accuracy and so on. The same is true of linear system solvers, other numerical algorithms and data products. In some complicated computational tasks, the possible situations are more challenging. For example, there are many different system modeling algorithms developed for different control systems such as ARMX or ARMAX systems, time variant or invariant systems, noisy or noiseless systems, linear or nonlinear systems, and so on. So in this case it is more difficult for keywords and ontologies to describe precisely the real functionality of the service provider's algorithms.

Keywords and ontologies cannot be defined and interpreted precisely enough to make brokering or matchmaking between grid services robust in a truly distributed, heterogeneous computing environment. Thus matching conflicts will exist between a client's requests and a service provider's responses. Some form of functional validation of computing resources will be required.

Functional validation means that a client presents to a prospective service provider a sequence of *challenges*. The service provider replies to these challenges with corresponding *answers*. Only after the client is satisfied that the service provider's answers are consistent with the client's expectations is an actual commitment made to using the service. This is especially important in mission critical applications. In fact we can find the same idea of functional validation in our daily lives. For example, a demo is often used to show the functionality of some software.

Our ongoing research on agent-based systems [9] has led us to the conclusion that brokering at the purely symbolic level will not be sufficient to implement truly distributed, heterogeneous multi-agent computing. Two steps are required before agents commit to each other:

1. Service *identification* and *location*;
2. Service *functional validation*.
3. *Commitment* to the service.

These steps are shown in Figure 2. Identification and location will be performed by ORB or matchmaker agents and is already an area of active research. However, functional validation of distributed components and agents is a new subject of research that is essential for the future success of truly heterogeneous, distributed computing grids.

3. Machine Learning Models of Functional Validation

Our approach to functional validation is to allow the client to challenge the service provider with some test cases, $x_1, x_2, \ldots, x_k$. The remote service provider offers corresponding responses/answers, $f_R(x_1), f_R(x_2), \ldots, f_R(x_k)$. The client may or may not have independent access to the correct responses/answers, $f_C(x_1), f_C(x_2), \ldots, f_C(x_k)$. Depending on the sequence of responses, the client may or may not commit to using (and therefore possibly paying for) the service provided.

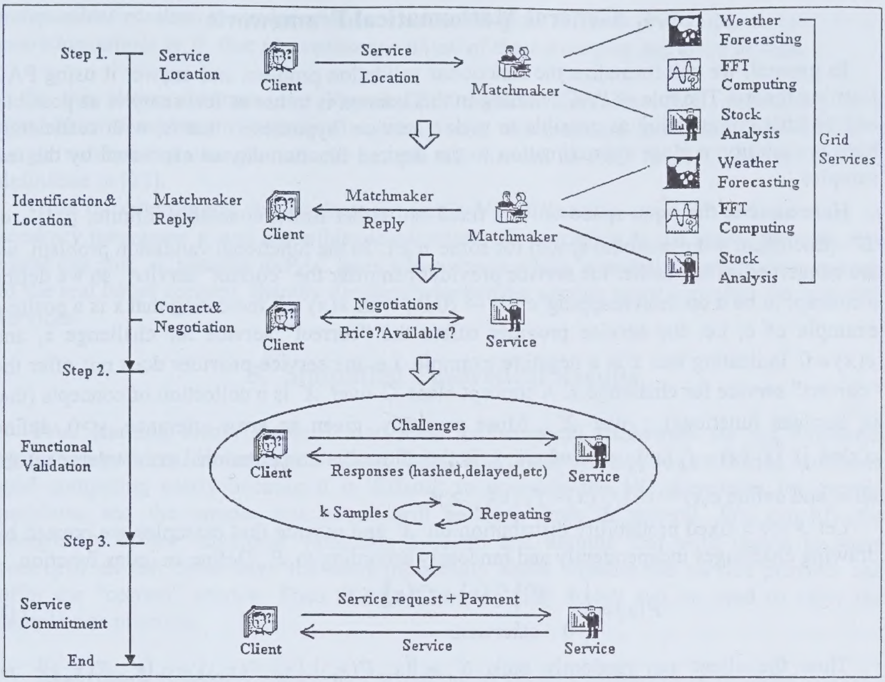

Step 1.
Service Location
Client
Service
Location
Matchmaker
Weather Forecasting
FFT Computing
Stock Analysis
Grid Services
Weather Forecasting
FFT Computing
Stock Analysis
Identification& Location
Matchmaker Reply
Client
Matchmaker
Reply
Matchmaker
Contact& Negotiation
Client
Negotiations
Price?Available?
Service
Step 2.
Functional Validation
Client
Challenges
Responses (hashed,delayed,etc)
Service
k Samples
Repeating
Step 3.
Service Commitment
Client
Service request + Payment
Service
Service
End

Figure 2: Grid services and functional validation

We will formalize the functional validation program for a computational and data service as follows. Denote the client's calling simulation by C and the remote service component by R. C requires the evaluation of a function, $f_C(x)$, where x is the input parameter. Assuming compatibility of input and output parameter structures and types, which has already been checked by the ORB or matchmaker, the remote service is expected to provide $f_R(x)$. Table 1 shows the possible situations that can arise in this problem [10].

Table 1: Possible situations arising in functional validation and their solutions

Client C	Server R	Machine Learning Model
"knows" $f_C(x)$	provides $f_R(x)$	PAC-learning and Chernoff bounds
"knows" $f_C(x)$	doesn't provide $f_R(x)$	Zero-knowledge proof
doesn't "know" $f_C(x)$	provides $f_R(x)$	Simulation-based and reinforcement learning

In this paper , we discuss the simplest case and assume that the client C itself has the correct value $f_C(x)$ for *the selected samples* and the candidate service provides responses $f_R(x)$ directly after C challenges R with the sample inputs. So the basic question in the functional validation problem is that in order to know whether the service provider can offer the "correct" service, how large should the sample size k be?

4. General Mathematical Framework

In general, we can formalize the functional validation problem and answer it using PAC learning theory. The role of PAC learning in this context is to use as few samples as possible, and as little computation as possible to pick a service (hypothesis) that is, with sufficiently high probability, a close approximation to the desired functionality as expressed by the test samples.

Here assume the input space X is a fixed set, either finite, countably infinite, $[0,1]^n$, or E^n (Euclidean n-dimensional space) for some $n \geq 1$. In the functional validation problem, we are concerned with whether the service provider can offer the "correct" service , so we define a concept to be a boolean mapping $c: X \to \{0,1\}$, with $c(x)=1$ indicating that x is a positive example of c, i.e. the service provider offers the "correct" service for challenge x, and $c(x)=0$ indicating that x is a negative example, i.e. the service provider does not offer the "correct" service for challenge x. A concept class C over X is a collection of concepts (that is, boolean functions) c over X. More precisely, given an error tolerance, $\gamma>0$, define $c(x)=1$ if $\|f_C(x)-f_R(x)\|_X \leq \gamma$ (where γ is the allowable computational error tolerance) for all x, and define $c(x)=0$ if $\|f_C(x)-f_R(x)\|_X \geq \gamma$.

Let P be a fixed probability distribution on X and assume that examples are created by drawing challenges independently and randomly according to P. Define an index function

$$F(x)=\begin{cases} 1 & \text{if } \|f_C(x)-f_R(x)\| \leq \gamma \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

Then the client can randomly pick $S_m = \{(x_1,F(x_1)),(x_2,F(x_2)),\cdots,(x_m,F(x_m))\}$ m samples to learn a hypothesis $h \in H$ about whether the service provider offers the "correct" service, where H is the hypothesis space and usually is the concept class C itself. Here let $A_{C,H}$ denotes the set of all learning functions $A: S_m \to H$. We claim that $A \in A_{C,H}$ is consistent if it agrees with the samples, that is $h = A(S_m)$. Thus based on the PAC learned hypothesis, the client can conclude whether the service provider can offer the "correct" service with the desired level of confidence.

Now consider the problem of how many samples or challenges are needed to make a decision about whether the hypothesis is a good enough approximation to the real target concept or not. Define the error between the target concept c and the hypothesis h as

$$error(h) = Prob_{x \in P}[c(x) \neq h(x)] \tag{2}$$

where $Prob_{x \in P}[.]$ indicates the probability with respect to the random drawing of x according to P. Then mathematically we can formalize the above problem as follows: How large must the number of challenges, m, be so that

$$Prob^m\{error(h) \leq \varepsilon\} \geq 1-\delta \tag{3}$$

where ε is the accuracy parameter and δ is the confidence parameter.

Blumer et. al. [11] solved this problem with the following powerful result:

Theorem 1 (Blumer et al.[11]) *Let H be any well-behaved hypothesis space of finite Vapnik-Chervonenkis dimension d contained in 2^X, P be any probability distribution on X and the target concept c be any Borel set contained in X. Then for any $0<\varepsilon,\delta<1$, given*

$$m \geq \max\left(\frac{4}{\varepsilon}\log\frac{2}{\delta}, \frac{8d}{\varepsilon}\log\frac{13}{\varepsilon}\right) \tag{4}$$

independent random examples of c drawn according to P*, with probability at least* $1-\delta$*, every hypothesis in* H *that is consistent with all of these examples has error at most* ε.

In the above theorem, the Vapnik-Chervonenkis dimension (VC dimension) is a combinatorial measure of concept class complexity which assigns to each concept class C a single number that characterizes the sample size needed to PAC learn C. See its detailed definition in [11].

Thus by determining the boolean concept's VC dimension over X and selecting an accuracy parameter ε and a confidence parameter δ, according to the above theorem , the client can pick m samples to PAC learn a hypothesis which is probabilistically close enough to the real target concept, thereby deciding whether the service agent can offer the "correct" services.

5. Simplified Theoretical Results

PAC learning theory formalizes an ideal mathematical framework for the functional validation problem. However, the above theoretical result can not be practically applied in grid computing easily because it is difficult to compute the VC dimension for generic problems and the sample size bound will be very large in general. We simplify the complexity of the functional validation problem and assume that with regard to some concepts, all test cases have the same probability about whether the service provider can offer the "correct" service. Then Chernoff bounds [12] theory can be used to solve the sample size problem.

Theorem 2 (Chernoff Bounds): *Consider independent identically distributed samples* $x_1, x_2, \cdots, x_m$ *from a Bernoulli distribution with expectation* p*. Define the empirical estimate of* p *based on these samples as*

$$\hat{p} = \frac{\sum_{i=1}^{m} x_i}{m} \tag{5}$$

Then for any $0 < \varepsilon, \delta < 1$*, if the sample size*

$$m \geq \frac{\ln\delta - \ln 2}{-2\varepsilon^2} \tag{6}$$

then the probability

$$prob^m\left[\left|\hat{p} - p\right| \geq \varepsilon\right] \leq \delta \tag{7}$$

During the functional validation process, the client only continues with the next sample if it knows that the service provider offers the "correct" response for all previous samples. Once the client encounters a negative example, i.e. the service provider offers a wrong reply for the current sample, the client will stop the validation process with this service provider and proceeds to negotiate with other possible service provider candidates. So with regard to this special situation, the client only needs to know how many samples are needed if all tested samples are positive examples (otherwise it just gives up). Thus the empirically estimated $\hat{p}$ should be equal to one. Then because $0 < p < 1$ and $\hat{p} = 1$, with the same sample size m defined in inequality (6), the inequality (7) can be updated by the following inequality:

$$prob^m\left[(1-p)\geq\varepsilon\right]\leq\frac{\delta}{2} \tag{8}$$

Then according to Theorem 2, it is straightforward to have the following corollary:

Corollary 2.1: *For the functional validation problem described above, given any* $0<\varepsilon,\delta<1$, *if the sample size*

$$m\geq\frac{\ln\delta}{-2\varepsilon^2} \tag{9}$$

then the probability

$$prob^m\left[(1-p)\geq\varepsilon\right]\leq\delta \tag{10}$$

In the above inequality, ε is the accuracy parameter which bounds the difference between the estimated probability and the true probability, while δ determines the confidence in this approximation. However, the goal of the functional validation process is to have a high confidence that the service will provide the correct response on the next request. Given a target probability, say *P*, the client agent needs to know how many consecutive positive samples, *m*, are required so that the next request to the service will be correct with probability *P*. We can combine the accuracy parameter, ε, and confidence, δ, together into this one parameter P. Using the definitions of δ and ε from above, it is easy to see that if the inequality

$$P\leq(1-\varepsilon)(1-\delta) \tag{11}$$

is satisfied, then the next sample will have probability P of being correct as well, given the Bernoulli model and other assumptions from above. Note that in inequality (9), ε and δ are both needed to compute the sample size m. So after a client agent selects a P, we still need to figure out how to compute the sample size m under the constraint inequality (11). We can formulate this problem as the following nonlinear optimization problem:

$$m=\min_{\varepsilon,\delta}f(\varepsilon,\delta)=\min_{\varepsilon,\delta}\left(\frac{\ln\delta}{-2\varepsilon^2}\right) \tag{12}$$

$$\text{s.t.: } (1-\varepsilon)(1-\delta)\geq P \text{ and } 0<\varepsilon,\delta<1.$$

The objective function $f(\varepsilon,\delta)$ has two variables. From the constraint inequality, we require

$$\delta\leq1-\frac{P}{1-\varepsilon} \tag{13}$$

Obviously when δ is larger, the sample size m is smaller. So if we choose $\delta=1-\frac{P}{1-\varepsilon}$ and think about the constraint $0<\varepsilon,1-\frac{P}{1-\varepsilon}<1$, we can transfer the above two-dimensional function optimization problem to the following one-dimensional function optimization problem:

$$m=\min_{\varepsilon}g(\varepsilon)=\min_{\varepsilon}\left(\frac{\ln(1-\frac{P}{1-\varepsilon})}{-2\varepsilon^2}\right) \tag{14}$$

$$\text{s.t. } 0<\varepsilon<1-P.$$

So when ε approaches the edge point 0 or $1-P$, the function $g(\varepsilon)$ will approach positive infinity. Unfortunately, we have not obtained a closed-form analytical solution because of the log function. However, we use elementary nonlinear function optimization methods to compute the minimum sample size, m, such as the steepest descent method. Figure 3 shows how the sample size changes with the given probability P.

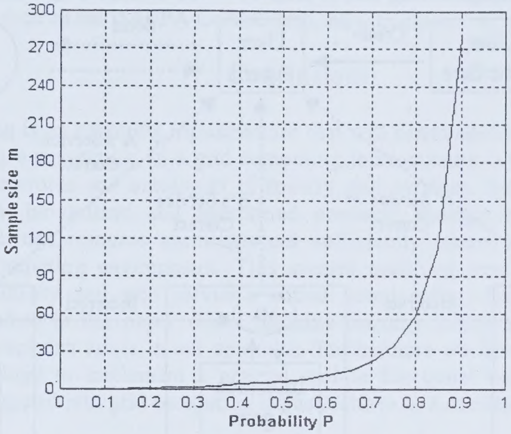


Figure 3: sample size vs. probability P curve

6. Mobile Functional Validation Agent

In addition to the above analytic results, we have implemented basic functional validation protocols for a grid computing infrastructure. We have implemented a test bed for the validation protocols, using the Dartmouth D'Agents system [9][13] to create mobile functional validation agents (MFVA) for grid computing.

D'Agents is a mobile-agent system developed at Dartmouth College. A mobile agent is a program that can migrate from machine to machine in a heterogeneous network. The program can choose when and where to migrate. It can suspend its execution at an arbitrary point, transport to another machine and resume execution on the new machine. Mobile agent have several advantages over the traditional client/server model: mobile agents consume fewer network resources and communicate efficiently; mobile agents do not require a continuous connection between machines; mobile agents hide the communication channels but not the location of the computation. So people can use mobile agent systems as a single, unified framework to implement a wide range of distributed applications easily and efficiently.

Consider the functional validation process for grid computing. In a traditional setting, during this process a client submits sample data, x, to the service provider and waits for the reply $f_R(x)$ from the remote service provider. For some difficult computing task, it may take the service provider a long time to compute the result. After the client gets the response $f_R(x)$, it must compare $f_C(x)$ with $f_R(x)$ to see whether the service provider offers an acceptable result for the sample. If $f_R(x)$ is acceptable, the client will send out the next sample data and repeat the above procedure. If $f_R(x)$ is unacceptable, the client will close the current communication channel and open a new one with another possible service candidate and then repeat the same validation process. So during this process, the client and the service provider can generate significant network traffic.

Using the D'Agents system, we have created a mobile functional validation agent to make the whole functional validation process more efficiently. Figure 4 shows how the mobile functional validation agent works in the grid computing infrastructure. After the ORB or matchmaker returns the machine list of possible service provider candidates, the client uses the simplified theoretical results in section 5 to compute the sample size *m* suitable for a desired probability of correctness. Then it creates a mobile agent that encapsulates the machine list, the *m* sample list and the client's actual computing request. The mobile functional validation agent jumps to the first machine that is listed on the machine list and executes the functional validation process on that local machine. During the validation process, if the mobile agent gets an unacceptable reply from the local service provider, it will autonomously give up this service provider, jump to the next machine on the machine list and initiate a new validation process on that machine. Otherwise it will continue to submit samples to the service provider and repeat the validation procedure. If it continuously gets correct responses for all *m* samples, it will ask for the client's actual computing service from the service provider. After it gets the result, it will autonomously jump back to the client machine and return the result to the client.

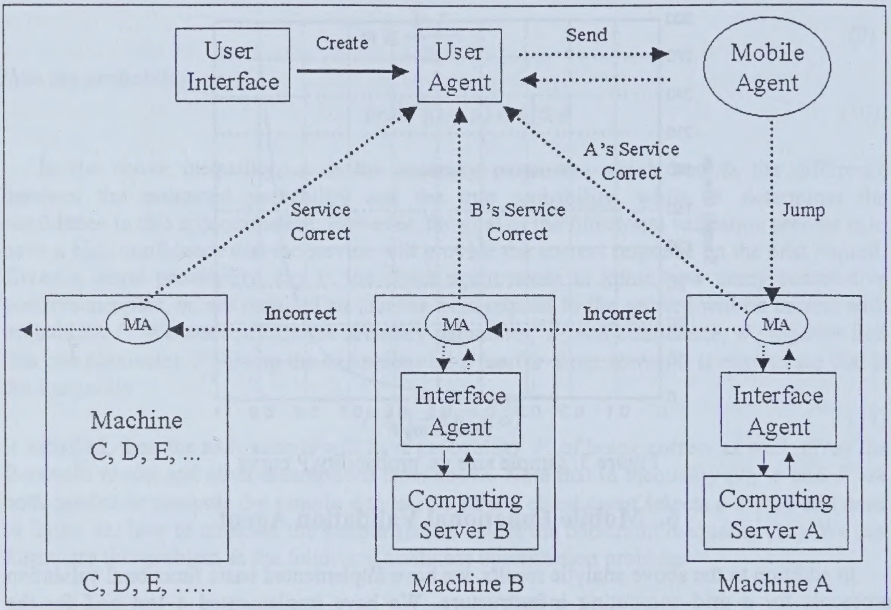

User Interface
Create
User Agent
Send
Mobile Agent
A's Service Correct
Service Correct
B's Service Correct
Jump
Incorrect
Incorrect
MA
MA
MA
Machine C, D, E, ...
Interface Agent
Computing Server B
Interface Agent
Computing Server A
C, D, E, ...
Machine B
Machine A

Figure 4: mobile functional validation agent

So using the mobile functional agent, the functional validation process is executed on the service provider machine and all communication is done between programs that are running on the same machine. This reduces network traffic and makes the validation process faster and more robust. Moreover, without the client's involvement, a mobile agent can autonomously complete the whole validation process and return the correct result to the client.

Because different service providers have different communication protocols, in Figure 4, interface agents are used to translate these protocols to a standard agent communication protocol. Our mobile functional validation agent system is a very general and standard system. For different computing tasks, a client only needs to change the sample lists in the user agent program and a separate result comparison class. So we believe this general mobile functional validation agent system can be integrated into grid computing infrastructures as a standard service, such as the DARPA CoABS grid infrastructure.

7. Conclusions

Grid computing is an emerging infrastructure that will fundamentally change the way we think about and use computing. In a grid computing infrastructure, a broker or matchmaker agent will use keywords and ontologies to specify grid services. However, keywords and ontologies cannot be defined and interpreted precisely enough to make brokering or matchmaking between resource agent services sufficiently robust in a truly distributed, heterogeneous computing environment. This creates matching conflicts between client's requested functionality and grid service's actual functionality. Functional validation is proposed and studied in this paper. Some machine learning theory is applied to solve the sample size problem that arises in our approach. Furthermore we have used the Dartmouth D'Agents technology to implement a general mobile functional validation agent system which can be integrated into grid computing infrastructures as a standard grid service.

References

[1] Foster, I., and Kesselman, C., 1998. *The Grid Blueprint for a New Computing Infrastructure*, Morgan Kaufmann Publishers, San Francisco, California.

[2] http://www.infospheres.caltech.edu

[3] http://coabs.globalinfotek.com

[4] Natan R.B.,1995. *Corba-A Guide to Common Object Request Broker Architecture*, McGraw-Hill, New York.

[5] Edwards, W. K. , 1999. *Core Jini*. Prentice Hall.

[6] http://logic.Stanford.edu/kif/specification.html

[7] http://www.cs.umbc.edu/kqml/

[8] Brigham, E.O., 1988. *The Fast Fourier Transform and Its Applications,* Prentice Hall, Englewood Cliffs, New Jersey.

[9] http://actcomm.dartmouth.edu and http://www.cs.dartmouth.edu/~agent/

[10] Cybenko, G. and Jiang, G., 1999. *Matching conflicts: Functional validation of agents.* 1999 AAAI workshop on Agents' Conflicts, pp. 14-19, Orlando, Florida.

[11] Blumer, A. et al., 1989. *Learnability and the Vapnik-Chervonenkis Dimension*. Journal of the Association for Computing Machinery 36(4):929.

[12] Kearns, M.J., and Vazirani, U.V., 1994. *An Introduction to Computational Learning Theory,* The MIT Press, Cambridge, Massachusetts.

[13] Kotz,D., Gray,R., Nog,S., Rus D., Chawla,S. and Cybenko,G. 1997. *Agent Tcl: Targeting the needs of mobile computers*. IEEE Internet Computing, 1(4):58.

Redundancy of the Lempel-Ziv codes

Serap A. Savari
Bell Laboratories, Lucent Technologies
Murray Hill, NJ 07928
savari@research.bell-labs.com

Abstract

The Lempel-Ziv codes are universal variable-to-fixed length codes that have become virtually standard in practical lossless data compression. For any given source output string from a Markov or unifilar source, we upper bound the difference between the number of binary digits needed to encode the string and the self-information of the string. We use this result to demonstrate that for unifilar, Markov sources, the redundancy of encoding the first n letters of the source output with the Lempel-Ziv string matching code (LZ '77) is $O((\ln\ln n)/(\ln n))$ and the redundancy with the Lempel-Ziv incremental parsing rule (LZ '78) is $O(1/(\ln n))$; in both cases, we upper bound the exact form of convergence.

1 Introduction

An important and challenging problem in data compression is trying to better understand the Lempel-Ziv codes (see [1] and [2]), which have motivated many practical lossless data compression schemes. We assume that we are encoding the output of a unifilar, Markov source. We define a Markov source, i.e., a unifilar source, with finite alphabet $\{0, 1, \ldots, K-1\}$ and set of states $\{0, 1, \ldots, R-1\}$ by specifying, for each state s and letter j,

1. the probability $p_{s,j}$ that the source emits j from state s and
2. the unique next state $S[s, j]$ after j is issued from state s.

For any source string σ with an initial state s_0, these rules inductively specify its final state $S[s_0, \sigma]$, its probability $P(\sigma|s_0)$, and its self-information in natural units, $I(\sigma|s_0) = -\ln P(\sigma|s_0)$. They also determine $\mathcal{H}$, the entropy of the source in natural units. The class of sources that can be modelled by a Markov source is fairly general.

We also assume that the output of the source is encoded into a string of binary digits from which the source output can be uniquely determined. The *redundancy* of a source code is the amount by which the average number of binary digits per source symbol produced by that code exceeds $\mathcal{H}\log_2 e$, the binary entropy of the source. In [3], Shannon demonstrated that the redundancy of any lossless data compression algorithm is non-negative and that there exists source codes for which the redundancy is arbitrarily close to zero.

Before we describe the Lempel-Ziv codes, we will introduce some notation. For any string ϕ and letter j, define the string $\phi \circ j$ as the string formed by appending j to the string ϕ. Let u_i denote the i^{th} output symbol and for $j \geq i$, let u_i^j represent the string $u_i \circ u_{i+1} \circ \cdots \circ u_j$.

We assume that we wish to encode the source output string u_1^n for some $n > 1$ and that the decoder knows in advance the length n of this string. We will first specify the Lempel-Ziv incremental parsing rule. For LZ '78, the next parsed phrase is the shortest prefix of the unparsed source output which has not appeared earlier as one of the parsed phrases. The encoder for LZ '78 uses and updates a dynamically expanding dictionary which contains all possibilities for the next parsed phrase. If the dictionary contains M entries, then $\lceil \log_2 M \rceil$ bits are used to encode the next parsed phrase.

For the version of LZ '77 we are considering, at each parsing point the next parsed phrase is the shortest prefix of the unparsed source output that does not appear earlier as a substring of the source output string. Let η denote the length of the parsed portion of the source output string. The binary representation for ϕ is an encoding of the length $l(\phi)$ of ϕ using $\log_2 l(\phi) + O(\ln \ln l(\phi))$ bits (e.g., see [4]), an encoding of the last letter of ϕ with $\lceil \log_2 K \rceil$ bits, and an encoding using $\lceil \log_2 \eta \rceil$ bits of where the first $l(\phi) - 1$ letters of ϕ have occurred earlier as a substring.

For $Alg \in \{\text{LZ '77, LZ '78}\}$, let $\mathcal{L}^{Alg}(u_1^n)$ denote the length of the encoding of the string u_1^n in bits for Alg. Let U_1^n be the random string corresponding to the first n letters emitted from the source. The redundancy $\mathcal{R}^{Alg}$ of the code Alg in bits is

$$\mathcal{R}^{Alg} = E\left(\frac{1}{n}\mathcal{L}^{Alg}(U_1^n)\right) - \mathcal{H} \log_2 e, \tag{1}$$

where the expectations are taken over all n-tuples. We will first discuss the literature on LZ '78. It was demonstrated in [2] that

$$\lim_{n\to\infty} \mathcal{R}^{\text{LZ '78}} = 0.$$

In [5], it was established that for a binary source, every source output string u_1^n satisfies

$$\frac{\mathcal{L}^{\text{LZ '78}}(u_1^n) - I(u_1^n|s_0)\log_2 e}{n} \leq \frac{\ln \ln n}{\ln n} + O\left(\frac{1}{\ln n}\right),$$

and hence,

$$\mathcal{R}^{\text{LZ '78}} \leq \frac{\ln \ln n}{\ln n} + O\left(\frac{1}{\ln n}\right).$$

In [5], it was conjectured that $\mathcal{R}^{\text{LZ '78}} = \Theta\left((\ln \ln n)/(\ln n)\right)$. However, in recent years, much of the data compression community believed that $\mathcal{R}^{\text{LZ '78}} = \Theta\left((\ln n)^{-1}\right)$. This question was next addressed in [6]; it was claimed there that for a binary, memoryless source, there exist a constant C which is a function of source parameters and a fluctuating function $\delta(n)$ with small amplitude that satisfies

$$\mathcal{R}^{\text{LZ '78}} = \frac{C + \delta(n)}{\ln n} + O\left(\frac{\ln \ln n}{(\ln n)^2}\right)$$

and that an extension of this result exists for those Markov sources for which each output symbol depends statistically only on the previous output symbol.

Next we will describe the literature on the version of LZ '77 that we are considering. The results of [5] suggest, but do not explicitly state, that for a binary source, every source output string u_1^n satisfies

$$\frac{\mathcal{L}^{\text{LZ '77}}(u_1^n) - I(u_1^n|s_0)\log_2 e}{n} \leq O\left(\frac{\ln \ln n}{\ln n}\right),$$

and hence,

$$\mathcal{R}^{\mathrm{LZ\ '77}} \le O\left(\frac{\ln\ln n}{\ln n}\right).$$

[7] considers the special case where each output symbol depends statistically only on the l previous symbols. If $\hat{I}_l(u_1^n)$ is the empirical entropy of l-tuples for the string u_1^n, then it is established in [7] that

$$\frac{\mathcal{L}^{\mathrm{LZ\ '77}}(u_1^n) - \hat{I}_l(u_1^n)\log_2 e}{n} \le O\left(\frac{\ln\ln n}{\ln n}\right).$$

Other versions of the Lempel-Ziv string matching code have been described and analyzed in [8], [9], and [10].

Our approach to analyzing the performance of an encoding rule is new. We show how to use renewal theory to bound the number of phrases corresponding to the parsing of a string in terms of the self-information of the string. This leads to an upper bound on the length of the encoding of the string. Our main result is to demonstrate that

$$\begin{aligned}
\frac{\mathcal{L}^{\mathrm{LZ\ '78}}(u_1^n) - I(u_1^n|s_0)\log_2 e}{n} &\le \frac{I(u_1^n|s_0)}{n}\log_2\left(\frac{R(K-1)\log_2 e}{\mathcal{H}}\right)\cdot\frac{1}{\ln n} + o\left(\frac{1}{\ln n}\right)\\
\frac{\mathcal{L}^{\mathrm{LZ\ '77}}(u_1^n) - I(u_1^n|s_0)\log_2 e}{n} &\le \frac{2I(u_1^n|s_0)\log_2 e}{n}\cdot\frac{\ln\ln n}{\ln n} + o\left(\frac{\ln\ln n}{\ln n}\right);
\end{aligned}$$

hence,

$$\begin{aligned}
\mathcal{R}^{\mathrm{LZ\ '78}} &\le \mathcal{H}\log_2\left(\frac{R(K-1)\log_2 e}{\mathcal{H}}\right)\cdot\frac{1}{\ln n} + o\left(\frac{1}{\ln n}\right)\\
\mathcal{R}^{\mathrm{LZ\ '77}} &\le 2\mathcal{H}\log_2 e\cdot\frac{\ln\ln n}{\ln n} + o\left(\frac{\ln\ln n}{\ln n}\right).
\end{aligned}$$

2 New Redundancy Bounds

In evaluating $\mathcal{L}^{\mathrm{LZ\ '78}}(u_1^n)$, we will use the following elementary result (see [11]).

Lemma 1 *For any integer $k \ge 2$, and real number $x \ge 0$,*

$$\sum_{i=1}^{k}\lceil x + \log_2 i\rceil \le k\log_2 k + kx + k\log_2\left(\frac{\log_2 e}{e}\right) + O(\ln k).$$

For $Alg \in \{\text{LZ '77}, \text{LZ '78}\}$, let c^{Alg} symbolize the number of phrases obtained by parsing u_1^n according to Alg. For LZ '78, the dictionary starts with K entries and has a net gain of $K-1$ symbols per parse; hence, the size of the dictionary used to select the ψ^{th} parsed string is $\psi(K-1)+1$. Since $\lceil\log_2 M\rceil$ bits are used to encode any entry of a dictionary of size M for each parsing rule, it follows from Lemma 1 that

$$\begin{aligned}
\mathcal{L}^{\mathrm{LZ\ '78}}(u_1^n) &= \sum_{j=1}^{c^{\mathrm{LZ\ '78}}}\lceil\log_2(j(K-1)+1)\rceil\\
&\le c^{\mathrm{LZ\ '78}}\log_2 c^{\mathrm{LZ\ '78}} + c^{\mathrm{LZ\ '78}}\log_2\left(\frac{(K-1)\log_2 e}{e}\right) + O\left(\ln c^{\mathrm{LZ\ '78}}\right). \quad (2)
\end{aligned}$$

We next bound $\mathcal{L}^{\text{LZ '77}}$ in terms of $c^{\text{LZ '77}}$. Let l_i denote the length of i^{th} parsed phrase for LZ '77. For each phrase the encoder requires at most $\lceil \log_2 n \rceil$ bits to encode the position at which the matched substring begins. Therefore, phrase i is encoded with at most $\lceil \log_2 n \rceil + \log_2 l_i + O(\ln \ln l_i) + \lceil \log_2 K \rceil + 1$ bits. Hence,

$$\begin{aligned}
\mathcal{L}^{\text{LZ '77}}(u_1^n) &\le \sum_{i=1}^{c^{\text{LZ '77}}} (\log_2 n + \log_2 l_i + O(\ln \ln l_i) + O(1)) \\
&= c^{\text{LZ '77}} \log_2 \left(c^{\text{LZ '77}} \cdot \frac{n}{c^{\text{LZ '77}}} \right) + c^{\text{LZ '77}} \sum_{i=1}^{c^{\text{LZ '77}}} \frac{\log_2 l_i + O(\ln \ln l_i) + O(1)}{c^{\text{LZ '77}}} \\
&\le c^{\text{LZ '77}} \log_2 c^{\text{LZ '77}} + 2c^{\text{LZ '77}} \log_2 \left(\frac{n}{c^{\text{LZ '77}}} \right) + c^{\text{LZ '77}} O \left(\ln \ln \left(\frac{n}{c^{\text{LZ '77}}} \right) \right) \qquad (3)
\end{aligned}$$

by Jensen's inequality.

For $Alg \in \{\text{LZ '78}, \text{LZ '77}\}$, we see from (2) and (3) that upper bounds on c^{Alg} lead to upper bounds on $\mathcal{L}^{Alg}$. For the remainder of the paper, we assume a source in which the self-information per symbol has a non-arithmetic distribution; i.e., there is no constant Λ such that $-\ln p_{s,j}$ is an integer multiple of Λ for all pairs of states s and symbols j satisfying $p_{s,j} > 0$. The situation in which the self-information corresponding to the source symbols issued has an *arithmetic* distribution with period Λ is treated in [11]. We have the following results.

Theorem 1 *Assume that the source has positive entropy $\mathcal{H}$. Abbreviate $I(u_1^n | s_0) < \infty$ by I. For $Alg \in \{LZ\ '77,\ LZ\ '78\}$, we have the following asymptotic relationships between the number of phrases associated with the parsing of u_1^n and the self-information of u_1^n.*

$$\frac{c^{Alg}}{n} \le \frac{I}{n} \cdot \frac{1}{\ln(\mathcal{H}I/R)} (1 + o(1)) = \frac{I}{n} \cdot \frac{1}{\ln n} (1 + o(1)) \qquad (4)$$

$$\frac{c^{Alg} \log_2 c^{Alg} - I \log_2 e}{n} \le \frac{I}{n} \cdot \frac{1}{\ln(\mathcal{H}I/R)} \left(\log_2 \left(\frac{Re}{\mathcal{H}} \right) + o(1) \right) = \frac{I}{n} \cdot \frac{1}{\ln n} \left(\log_2 \left(\frac{Re}{\mathcal{H}} \right) + o(1) \right). \qquad (5)$$

The asymptotic relationship between the length of the encoding of u_1^n and the self-information of u_1^n satisfies

$$\frac{\mathcal{L}^{\text{LZ '78}}(u_1^n) - I \log_2 e}{n} \le \frac{I}{n} \log_2 \left(\frac{R(K-1) \log_2 e}{\mathcal{H}} \right) \cdot \frac{1}{\ln n} + o \left(\frac{1}{\ln n} \right) \qquad (6)$$

$$\frac{\mathcal{L}^{\text{LZ '77}}(u_1^n) - I \log_2 e}{n} \le \frac{2I \log_2 e}{n} \cdot \frac{\ln \ln n}{\ln n} + o \left(\frac{\ln \ln n}{\ln n} \right). \qquad (7)$$

Proof: We introduce the following notation for Alg, $Alg \in \{\text{LZ '77}, \text{LZ '78}\}$. Let

- σ_i^{Alg} denote the i^{th} phrase of the source output
- ψ_i^{Alg} denote the source state just before phrase σ_i^{Alg}
- $\Omega = \{\tau : I(\sigma | s) = \tau$ for some string σ and state $s\}$
- $c^{Alg}(\tau) = |\{\sigma_i^{Alg} : 1 \le i \le c^{Alg} - 1,\ I(\sigma_i^{Alg} | \psi_i^{Alg}) = \tau\}|$

- $\mu(\tau) = |\{(s,\sigma) : I(\sigma|s) = \tau\}|$
- $\gamma(\tau) = |\{(s,\sigma) : I(\sigma|s) < \tau\}|$
- $\mathcal{I}^{Alg} = \sum_{i=1}^{c^{Alg}-1} I(\sigma_i^{Alg}|\psi_i^{Alg})$; $\mathcal{I}^{Alg} \leq I$.

Note that

$$c^{Alg} - 1 = \sum_{\tau \in \Omega} c^{Alg}(\tau) \tag{8}$$

and

$$\mathcal{I}^{Alg} = \sum_{\tau \in \Omega} \tau \cdot c^{Alg}(\tau). \tag{9}$$

To upper bound c^{Alg} for a given $I(u_1^n|s_0)$, we maximize $1 + \sum_{\tau\in\Omega} c^{Alg}(\tau)$ subject to the constraints $\sum_{\tau\in\Omega} \tau \cdot c^{Alg}(\tau) \leq I$ and $0 \leq c^{Alg}(\tau) \leq \mu(\tau)$ for all $\tau \in \Omega$. We will show that the number of phrases is maximized by selecting as many phrases with small self-information as possible. In particular, we have the following result.

Lemma 2 *For $Alg \in \{LZ\ '77,\ LZ\ '78\}$, an upper bound on c^{Alg} is given by*

$$c^{Alg} \leq \rho \doteq \gamma(\overline{\tau}) + \alpha \tag{10}$$

where $\overline{\tau}$ and α are chosen so that

$$\mathcal{J} \doteq \sum_{\tau\in\Omega:\tau<\overline{\tau}} \tau \cdot \mu(\tau) + \overline{\tau} \cdot \alpha > I \geq \mathcal{J} - \overline{\tau} \tag{11}$$

$$1 \leq \alpha \leq \mu(\overline{\tau}). \tag{12}$$

Note that $\overline{\tau}$ is a non-decreasing function of I and that as n approaches infinity, both I and $\overline{\tau}$ approach infinity.

Proof of Lemma 2: To arrive at a contradiction, suppose that (10) is false. Then $c^{Alg} - 1 \geq \rho$. The encoding rule ensures that each one of the first $c^{Alg} - 1$ phrases in the parsing is distinct. Let h_i^{Alg} denote the self-information of phrase i under parsing rule Alg; i.e., $h_i^{Alg} = I(\sigma_i^{Alg}|\psi_i^{Alg})$. Without loss of generality, reorder the self-informations so that $h_1^{Alg} \leq h_2^{Alg} \leq \cdots \leq h_{c^{Alg}-1}^{Alg}$. Now consider the $\gamma(\overline{\tau})$ distinct pairs (s,ϕ) with $I(\phi|s) < \overline{\tau}$ and choose α distinct pairs (s,ϕ) with $I(\phi|s) = \overline{\tau}$. Order the ρ pairs so that $I(\phi_1|s_1) \leq I(\phi_2|s_2) \leq \cdots \leq I(\phi_\rho|s_\rho)$. Observe that for each $i \in \{1,2,\ldots,\rho\}$, $h_i^{Alg} \geq I(\phi_i|s_i)$. Hence,

$$I \geq \mathcal{I}^{Alg} = \sum_{i=1}^{c^{Alg}-1} h_i^{Alg} \geq \sum_{i=1}^{\rho} h_i^{Alg} \geq \sum_{i=1}^{\rho} I(\phi_i|s_i) = \mathcal{J} > I,$$

which is a contradiction. □

To understand the relationships among $\overline{\tau}$, ρ, and I, we investigate how the source generates self-information. We can model the generation of self-information as a *renewal process* (see [12, §3] and [13, §3]). In a renewal process, renewals occur at randomly chosen epochs in time and successive interrenewal periods, i.e., the intervals between renewals, are independent and identically distributed non-negative random variables; we assume that the process starts evolving at time zero. A related type of stochastic process is a *delayed renewal process.* A delayed renewal process is almost identical to an ordinary renewal process; the only difference is that the first interrenewal period of a delayed renewal process may have a different probability distribution from the remaining ones. For our purposes, we choose the interrenewal periods to represent the self-information generated by the source between successive returns to some given state ψ; an epoch can then be interpreted as the self-information of the source string upon an entrance of the source into state ψ. Because of the Markovian nature of the source, these interrenewal periods are independent random variables and all but the first interrenewal period are also identically distributed. That is, if the source is initially in state ψ, the stochastic process defined above is a renewal process; otherwise, it is a delayed renewal process. For each state ψ and integer $k \geq 2$, we let $J_k^{(\psi)}$ symbolize the self-information, in natural units, generated by the source between the $k-1^{\text{st}}$ and the k^{th} occurrences of state ψ; $J_1^{(\psi)}$ denotes the self-information produced until the source reaches state ψ for the first time. Let $T_k^{(\psi)} = J_1^{(\psi)} + \ldots + J_k^{(\psi)}$, and let $\{N^{(\psi,s)}(t); t \geq 0\}$ be the renewal or delayed renewal process defined, for each state ψ, and starting state s, by specifying the random variable $N^{(\psi,s)}(t)$ as the number of renewals until the self-information reaches t, i.e., the largest integer k for which $T_k^{(\psi)} \leq t < T_{k+1}^{(\psi)}$. Let $m^{(\psi,s)}(t) = E[N^{(\psi,s)}(t)]$. Given the starting state s and the source output $v_1, v_2, \ldots$, each prefix v_1^i with $S[s, v_1^i] = \psi$ corresponds to a renewal in the corresponding sample function; the expected number of renewals in the interval $(t, t+dt]$ is given by

$$m^{(\psi,s)}(t+dt) - m^{(\psi,s)}(t) = \sum_{\sigma: I(\sigma|s) \in (t,t+dt],\ \psi = S[s,\sigma]} P(\sigma|s). \tag{13}$$

For each string σ with $I(\sigma|s) \in (t, t+dt]$, we have that $e^{-t-dt} \leq P(\sigma|s) < e^{-t}$. Hence, it follows from (13) that

$$\begin{aligned} e^{-t-dt} \cdot \mid \sigma : I(\sigma|s) \in (t,t+dt],\ \psi = S[s,\sigma] \mid \ &\leq\ m^{(\psi,s)}(t+dt) - m^{(\psi,s)}(t) \\ &<\ e^{-t} \cdot \mid \sigma : I(\sigma|s) \in (t,t+dt],\ \psi = S[s,\sigma] \mid, \end{aligned}$$

and thus, the number of strings σ with $I(\sigma|s) \in (t, t+dt]$ and $\psi = S[s,\sigma]$ is in the interval $(e^t[m^{(\psi,s)}(t+dt) - m^{(\psi,s)}(t)], e^{t+dt}[m^{(\psi,s)}(t+dt) - m^{(\psi,s)}(t)]]$. It follows from (10) that

$$\sum_{s=0}^{R-1} \sum_{\psi=0}^{R-1} \int_0^{\bar{\tau}^-} e^x \, dm^{(\psi,s)}(x) \ < \ \rho \ \leq \ \sum_{s=0}^{R-1} \sum_{\psi=0}^{R-1} \int_0^{\bar{\tau}^+} e^x \, dm^{(\psi,s)}(x). \tag{14}$$

(10) and (11) imply that

$$\mathcal{J} \ = \sum_{\tau \in \Omega : \tau < \bar{\tau}} (\tau - \bar{\tau}) \mu(\tau) + \bar{\tau} \rho, \tag{15}$$

and thus,

$$\mathcal{J} \ = \ \sum_{s=0}^{R-1} \sum_{\psi=0}^{R-1} \int_0^{\bar{\tau}^-} (x - \bar{\tau}) e^x \, dm^{(\psi,s)}(x) + \bar{\tau} \rho. \tag{16}$$

Hence, by multiplying both sides of (14) and (16) by $e^{-\bar{\tau}}$, we find that

$$\sum_{s=0}^{R-1} \sum_{\psi=0}^{R-1} \int_0^{\bar{\tau}^-} e^{x-\bar{\tau}} \, dm^{(\psi,s)}(x) \ < \ \rho \cdot e^{-\bar{\tau}} \ \leq \ \sum_{s=0}^{R-1} \sum_{\psi=0}^{R-1} \int_0^{\bar{\tau}^+} e^{x-\bar{\tau}} \, dm^{(\psi,s)}(x) \tag{17}$$

$$\mathcal{J} \cdot e^{-\bar{\tau}} \ = \sum_{s=0}^{R-1} \sum_{\psi=0}^{R-1} \int_0^{\bar{\tau}^-} (x - \bar{\tau}) e^{x-\bar{\tau}} \, dm^{(\psi,s)}(x) + \bar{\tau} \rho \cdot e^{-\bar{\tau}}. \tag{18}$$

In [11], we establish the following result.

Lemma 3 *For all states s,*

$$\lim_{\overline{\tau}\to\infty}\int_0^{\overline{\tau}^-} e^{x-\overline{\tau}}\;dm^{(\psi,s)}(x) \;=\; \lim_{\overline{\tau}\to\infty}\int_0^{\overline{\tau}^+} e^{x-\overline{\tau}}\;dm^{(\psi,s)}(x) \;=\; \frac{\pi_\psi}{\mathcal{H}} \tag{19}$$

$$\lim_{\overline{\tau}\to\infty}\int_0^{\overline{\tau}^-} (x-\overline{\tau})e^{x-\overline{\tau}}\;dm^{(\psi,s)}(x) \;=\; -\frac{\pi_\psi}{\mathcal{H}}. \tag{20}$$

It follows from (17), (18) and Lemma 3 that as $\overline{\tau}$ increases,

$$\rho\cdot e^{-\overline{\tau}} \;=\; \frac{R}{\mathcal{H}}+o(1) \tag{21}$$

$$(\overline{\tau}\rho-\mathcal{J})\cdot e^{-\overline{\tau}} \;=\; \frac{R}{\mathcal{H}}+o(1). \tag{22}$$

Since $\mathcal{J}-\overline{\tau}\le I<\mathcal{J}$, (22) implies that as $\overline{\tau}$ increases,

$$(\overline{\tau}\rho-I)\cdot e^{-\overline{\tau}} \;=\; \frac{R}{\mathcal{H}}+o(1). \tag{23}$$

Taking the logarithm of both sides of (21), we find that as $\overline{\tau}$ increases,

$$\log_2\rho-\overline{\tau}\log_2 e \;=\; \log_2\left(\frac{R}{\mathcal{H}}\right)+o(1). \tag{24}$$

Multiplying (21) by (24), we see that

$$(\rho\log_2\rho-\overline{\tau}\rho\log_2 e)\cdot e^{-\overline{\tau}} \;=\; \frac{R}{\mathcal{H}}\log_2\left(\frac{R}{\mathcal{H}}\right)+o(1). \tag{25}$$

Multiplying both sides of (23) by $\log_2 e$ and adding the resulting expression to (25), we find that as $\overline{\tau}$ increases

$$(\rho\log_2\rho-I\log_2 e)\cdot e^{-\overline{\tau}} \;=\; \frac{R}{\mathcal{H}}\log_2\left(\frac{Re}{\mathcal{H}}\right)+o(1). \tag{26}$$

Next, we would like to determine the asymptotic relationship between $e^{-\overline{\tau}}$ and I. Substituting (21) into (23), we see that

$$\overline{\tau}\frac{R}{\mathcal{H}}+o(\overline{\tau})-I\cdot e^{-\overline{\tau}} \;=\; 0. \tag{27}$$

Define δ to satisfy

$$\overline{\tau} \;=\; \ln\left(\frac{\frac{\mathcal{H}I}{R}}{\ln\left(\frac{\mathcal{H}I}{R}\right)}(1+\delta)\right). \tag{28}$$

Substituting (28) into (27), we find that $\delta=o(1)$ and hence, as $\overline{\tau}$ and I increase,

$$e^{-\overline{\tau}} \;=\; \left(\frac{R}{\mathcal{H}I}\ln\left(\frac{\mathcal{H}I}{R}\right)\right)(1+o(1)). \tag{29}$$

Substituting (29) into (21) and (26), we see that as $\overline{\tau}$ and I increase,

$$\rho\cdot\left(\frac{R}{\mathcal{H}I}\ln\left(\frac{\mathcal{H}I}{R}\right)\right)(1+o(1)) \;=\; \frac{R}{\mathcal{H}}+o(1), \tag{30}$$

$$(\rho\log_2\rho-I\log_2 e)\cdot\left(\frac{R}{\mathcal{H}I}\ln\left(\frac{\mathcal{H}I}{R}\right)\right)(1+o(1)) \;=\; \frac{R}{\mathcal{H}}\log_2\left(\frac{Re}{\mathcal{H}}\right)+o(1). \tag{31}$$

By Lemma 2, $c^{Alg} \le \rho$. Hence, by (30) and (31), as I increases,

$$c^{Alg}\cdot\left(\frac{R}{\mathcal{H}I}\ln\left(\frac{\mathcal{H}I}{R}\right)\right)(1+o(1)) \;\le\; \frac{R}{\mathcal{H}}+o(1) \tag{32}$$

$$(c^{Alg}\log_2 c^{Alg} - I\log_2 e)\cdot\left(\frac{R}{\mathcal{H}I}\ln\left(\frac{\mathcal{H}I}{R}\right)\right)(1+o(1)) \;\le\; \frac{R}{\mathcal{H}}\log_2\left(\frac{Re}{\mathcal{H}}\right)+o(1). \tag{33}$$

Since the source has positive entropy, $I = \Theta(n)$, and therefore, (32) and (33) are equivalent to (4) and (5), respectively. (6) follows from (2), (4), and (5). Finally, we demonstrate (7). (3), (4), and (5) imply that as n increases,

$$\begin{aligned}
\frac{\mathcal{L}^{\text{LZ '77}}(u_1^n) - I\log_2 e}{n} &\le \frac{2c^{\text{LZ '77}}}{n}\log_2\Big(\frac{n}{c^{\text{LZ '77}}}\Big) + \frac{c^{\text{LZ '77}}}{n}O\Big(\ln\ln\Big(\frac{n}{c^{\text{LZ '77}}}\Big)\Big) \\
&\quad + \frac{c^{\text{LZ '77}}\log_2 c^{\text{LZ '77}} - I\log_2 e}{n} \\
&\le \frac{2c^{\text{LZ '77}}}{n}\log_2\Big(\frac{n}{c^{\text{LZ '77}}}\Big) + \frac{c^{\text{LZ '77}}}{n}O\Big(\ln\ln\Big(\frac{n}{c^{\text{LZ '77}}}\Big)\Big) + O\left(\frac{1}{\ln n}\right)
\end{aligned} \tag{34}$$

Over the interval $x \in (0, e^{-1})$, the functions $2x\log_2(x^{-1})$ and $x\ln\ln(x^{-1})$ are increasing. Hence, for sufficiently large n, the right-hand side of (34) is maximized at the upper bound on $c^{\text{LZ '77}}/n$ provided in (4), and this leads to (7). □

For a very probable collection of strings, we have the following result.

Corollary 1 *Assume that the source has positive entropy. With probability* $1 - O\left(\frac{1}{\sqrt{n}}\right)$,

$$\begin{aligned}
\frac{\mathcal{L}^{\text{LZ '78}}(U_1^n) - I(U_1^n|s_0)\log_2 e}{n} &\le \frac{\mathcal{H}}{\ln n}\left(\log_2\left(\frac{R(K-1)\log_2 e}{\mathcal{H}}\right)\right) + o\left(\frac{1}{\ln n}\right) \\
\frac{\mathcal{L}^{\text{LZ '77}}(U_1^n) - I(U_1^n|s_0)\log_2 e}{n} &\le 2\mathcal{H}\log_2 e\cdot\frac{\ln\ln n}{\ln n} + o\left(\frac{\ln\ln n}{\ln n}\right).
\end{aligned}$$

Proof: Since $\lim_{n\to\infty} E\left(I(U_1^n|s_0)/n\right) = \mathcal{H}$, and the variance and third moment of the self-information of every source symbol emitted is finite, it follows from [14] that

$$\mathcal{H} - o\left(\frac{1}{\ln n}\right) \le \frac{I(U_1^n|s_0)}{n} \le \mathcal{H} + o\left(\frac{1}{\ln n}\right) \quad \text{with probability } 1 - O\left(\tfrac{1}{\sqrt{n}}\right). \tag{35}$$

The corollary follows from (35), (6), and (7). □

It is also easy to derive upper bounds on the redundancy of the codes using Theorem 1 and (35). We prove the following results in [11] and [15].

Theorem 2 *Assume the source has positive entropy. Then*

$$\begin{aligned}
\mathcal{R}^{\text{LZ '78}} &\le \frac{\mathcal{H}}{\ln n}\left(\log_2\left(\frac{R(K-1)\log_2 e}{\mathcal{H}}\right)\right) + o\left(\frac{1}{\ln n}\right) \\
\mathcal{R}^{\text{LZ '77}} &\le 2\mathcal{H}\log_2 e\cdot\frac{\ln\ln n}{\ln n} + o\left(\frac{\ln\ln n}{\ln n}\right).
\end{aligned}$$

Theorems 1 and 2 suggest that LZ '78 asymptotically outperforms the version of LZ '77 that we have been considering. The reason for this is that for very long source output strings, the encoding of the phrase lengths for LZ '77 more than offsets the reduction in the number of phrases over LZ '78. This result is somewhat surprising since recent trends in practical lossless data compression suggest that LZ '77 and its variants perform as well as, if not better than, LZ '78 and its variants. There are a few explanations for this. For small to moderate values of n, the difference between $\Theta\left((\ln \ln n)/(\ln n)\right)$ and $\Theta\left((\ln n)^{-1}\right)$ is not significant and an understanding of the lower order terms is very important in order to make a fair comparison. The other limitations to our analysis are the assumptions that the source statistics do not change over time and that the dictionary of source strings can grow arbitrarily large. If either of these assumptions are violated, then the situation may be very different.

Thus far, we have been assuming that the source has positive entropy. To finish our analysis of the Lempel-Ziv codes, we will summarize our findings on the performance of the algorithms on the output of a source with *zero* entropy. The details of the analysis can be found in [11] and [15].

Theorem 3 *Assume the source has zero entropy. We have the following asymptotic bounds on the length of the encoding of the source output string* u_1^n.

$$\frac{\mathcal{L}^{\mathrm{LZ\ '78}}(u_1^n)}{n} \leq \sqrt{\frac{R}{2n}}\log_2 n + \sqrt{\frac{2R}{n}}\log_2\left(\frac{\sqrt{2R}(K-1)\log_2 e}{e}\right) + O\left(\frac{\ln n}{n}\right) \tag{36}$$

$$\frac{\mathcal{L}^{\mathrm{LZ\ '77}}(u_1^n)}{n} \leq \frac{2(R+1)\log_2 n}{n} + O\left(\frac{R \ln \ln n}{n}\right). \tag{37}$$

3 Conclusions

We have investigated the redundancy of LZ '78 and a version of LZ '77 as the length n of the source string encoded tends toward infinity. We established that the redundancy of LZ '78 and LZ '77 decrease towards zero as $O\left(1/(\ln n)\right)$ and $O\left((\ln \ln n)/(\ln n)\right)$, respectively. Moreover, we upper bounded the number of code symbols per source symbol needed for the encoding of a particular string u_1^n. The main idea was to upper bound the number of parsed phrases used to encode u_1^n by the maximum number of phrases with cumulative self-information less than $I(u_1^n|s_0)$ and to use renewal theory to understand the asymptotic behavior of the second quantity.

4 Acknowledgment

The results on LZ '78 were from a chapter of [16], which was supervised by Professor Robert G. Gallager, who suggested using renewal theory in analyzing source codes. The author is grateful to Professor Gallager and Professor Meir Feder for many helpful discussions and comments.

References

[1] J. Ziv and A. Lempel, "A universal algorithm for data compression," *I.E.E.E. Trans. Inform. Theory* IT-23, 337-343, 1977.

[2] J. Ziv and A. Lempel, "Compression of individual sequences via variable-rate coding," *I.E.E.E. Trans. Inform. Theory* IT-24, 530-536, 1978.

[3] C. E. Shannon, "A mathematical theory of communication," *Bell System Tech. J.* 27, 379-423, 623-656, 1948.

[4] P. Elias, "Universal codeword sets and representations of the integers," *I.E.E.E. Trans. Inform. Theory* IT-21, 194-203, 1975.

[5] E. Plotnik, M. J. Weinberger, and J. Ziv, "Upper bounds on the probability of sequences emitted by finite-state sources and on the redundancy of the Lempel-Ziv algorithm," *I.E.E.E. Trans. Inform. Theory,* IT-38, 66-72, 1992.

[6] G. Louchard and W. Szpankowski, "On the average redundancy rate of the Lempel-Ziv code," *Proceedings of the 1996 IEEE Data Compression Conference,* Snowbird, Utah, 1996.

[7] S. R. Kosaraju and G. Manzini, "Some entropic bounds for Lempel-Ziv algorithms," Unpublished manuscript.

[8] A. J. Wyner, *String Matching Theorems and Applications to Data Compression and Statistics,* Ph.D. thesis, Stanford University, 1993.

[9] A. D. Wyner and J. Ziv, "The sliding-window Lempel-Ziv algorithm is asymptotically optimal," *Proc. I.E.E.E.* 82, 872-877, 1994.

[10] A. D. Wyner and A. J. Wyner, "Improved redundancy of a version of the Lempel-Ziv algorithm," *I.E.E.E. Trans. Inform. Theory* IT-41, 723-731, 1995.

[11] S. A. Savari, "Redundancy of the Lempel-Ziv incremental parsing rule," *I.E.E.E. Trans. Inform. Theory* 43, 9-21, 1997.

[12] S. M. Ross, *Stochastic Processes,* Wiley, New York 1983.

[13] R. G. Gallager, *Discrete Stochastic Processes,* Kluwer, Boston 1996.

[14] B. V. Gnedenko and A. N. Kolmogorov, *Limit Distributions for Sums of Independent Random Variables,* Addison-Wesley Pub. Co., Cambridge, MA, 1968.

[15] S. A. Savari, "Redundancy of the Lempel-Ziv string matching code," *I.E.E.E. Trans. Inform. Theory,* 44, 787-791.

[16] S. A. Savari, *Variable-to-Fixed Length Codes for Sources with Known and Unknown Memory,* Ph.D. thesis, Massachusetts Institute of Technology, 1996.

The Information Bottleneck Method [*]

Naftali Tishby
The Hebrew University
Jerusalem 91904, Israel
tishby@cs.huji.ac.il

Fernando C. Pereira
AT&T Labs – Research
Florham Park, NJ 07932
pereira@research.att.com

William Bialek
NEC Research Institute
Princeton, NJ 08540
bialek@research.nj.nec.com

Abstract

We define the relevant information in a signal $x \in X$ as being the information that this signal provides about another signal $y \in Y$. Examples include the information that face images provide about the names of the people portrayed, or the information that speech sounds provide about the words spoken. Understanding the signal x requires more than just predicting y, it also requires specifying which features of X play a role in the prediction. We formalize the problem as that of finding a short code for X that preserves the maximum information about Y. That is, we squeeze the information that X provides about Y through a 'bottleneck' formed by a limited set of codewords $\tilde{X}$. This constrained optimization problem can be seen as a generalization of rate distortion theory in which the distortion measure $d(x, \tilde{x})$ emerges from the joint statistics of X and Y. The approach yields an exact set of self-consistent equations for the coding rules $X \to \tilde{X}$ and $\tilde{X} \to Y$. Solutions to these equations can be found by a convergent re-estimation method that generalizes the Blahut–Arimoto algorithm. Our variational principle provides a surprisingly rich framework for discussing a variety of problems in signal processing and learning, as will be described in detail elsewhere.

1 Introduction

A fundamental problem in formalizing our intuitive ideas about information is to provide a quantitative notion of "meaningful" or "relevant" information. These issues were intentionally left out of information theory in its original formulation by Shannon, who focused attention on the problem of transmitting information rather than judging its value to the recipient. Correspondingly, information theory has often been viewed as being strictly a theory of communication, and this view has become so accepted that many people consider statistical and information theoretic principles as almost irrelevant for the question of meaning. In contrast, we argue here that information theory, in particular lossy source compression, provides a natural quantitative approach to the question of "relevant information." Specifically, we formulate a variational principle for the extraction or efficient representation of relevant information. In related work [4] we argue that this single information theoretic principle contains as special cases a wide variety of problems, including prediction, filtering, and learning in its various forms.

---

[*] This research is supported in part by a grant from the US-Israel Binational Science Foundation (BSF).

The problem of extracting a relevant summary of data, a compressed description that captures only the relevant or meaningful information, is not well-posed without a suitable definition of relevance. A typical example is that of speech compression. One can consider lossless compression, but in any compression beyond the entropy of speech some components of the signal can not be reconstructed. On the other hand, a transcript of the spoken words has much lower entropy (by orders of magnitude) than the acoustic waveform, which means that it is possible to compress (much) further without losing any information about the words and their meaning.

The standard analysis of lossy source compression is "rate distortion theory," which provides a tradeoff between the rate, or signal representation size, and the average distortion of the reconstructed signal. Rate distortion theory determines the level of inevitable expected distortion, D, given the desired information rate, R, in terms of the *rate distortion function* $R(D)$. The main problem with rate distortion theory is in the need to specify the distortion function first, which in turn determines the relevant features of the signal. Those features, however, are often not explicitly known and an arbitrary choice of the distortion function is in fact an arbitrary feature selection.

In the speech example, we have at best very partial knowledge of what precise components of the signal are perceived by our (neural) speech recognition system. Those relevant components depend not only on the complex structure of the auditory nervous system, but also on the sounds and utterances to which we are exposed during our early life. It therefore is virtually impossible to come up with the "correct" distortion function for acoustic signals. The same type of difficulty exists in many similar problems, such as natural language processing, bioinformatics (for example, what features of protein sequences determine their structure) or neural coding (what information is encoded by spike trains and how). This is the fundamental problem of feature selection in pattern recognition. Rate distortion theory does not provide a full answer to this problem since the choice of the distortion function, which determines the relevant features, is not part of the theory. It is, however, a step in the right direction.

A possible solution comes from the fact that in many interesting cases we have access to an additional variable that determines what is relevant. In the speech case it might be the transcription of the signal, if we are interested in the speech recognition problem, or it might be the speaker's identity if speaker identification is our goal. For natural language processing, it might be the part-of-speech labels for words in grammar checking, but the dictionary senses of ambiguous words in information retrieval. Similarly, for the protein folding problem we have a joint database of sequences and three dimensional structures, and for neural coding a simultaneous recording of sensory stimuli and neural responses defines implicitly the relevant variables in each domain. All of these problems have the same formal underlying structure: extract the information from one variable that is relevant for the prediction of another one. The choice of additional variable determines the relevant components or features of the signal in each case.

In this short paper we formalize this intuitive idea using an information theoretic approach which extends elements of rate distortion theory. We derive self consistent equations and an iterative algorithm for finding representations of the signal that capture its relevant structure, and prove convergence of this algorithm.

2 Relevant quantization

Let X denote the signal (message) space with a fixed probability measure $p(x)$, and let $\tilde{X}$ denote its quantized codebook or compressed representation. For ease of exposition we assume here that both of these sets are finite, that is, a continuous space should first be quantized.

For each value $x \in X$ we seek a possibly stochastic mapping to a representative, or codeword in a codebook, $\tilde{x} \in \tilde{X}$, characterized by a conditional p.d.f. $p(\tilde{x}|x)$. The mapping $p(\tilde{x}|x)$ induces a soft partitioning of X in which each block is associated with one of the codebook elements $\tilde{x} \in \tilde{X}$, with probability given by

$$p(\tilde{x}) = \sum_x p(x)p(\tilde{x}|x) \ . \tag{1}$$

The average volume of the elements of X that are mapped to the same codeword is $2^{H(X|\tilde{X})}$, where

$$H(X|\tilde{X}) = -\sum_{x\in X} p(x) \sum_{\tilde{x}\in\tilde{X}} p(\tilde{x}|x)\log p(\tilde{x}|x) \tag{2}$$

is the conditional entropy of X given $\tilde{X}$.

What determines the quality of a quantization? The first factor is of course the rate, or the average number of bits per message needed to specify an element in the codebook without confusion. This number *per element of* X is bounded from below by the mutual information

$$I(X;\tilde{X}) = \sum_{x\in X}\sum_{\tilde{x}\in\tilde{X}} p(x,\tilde{x})\log\left[\frac{p(\tilde{x}|x)}{p(\tilde{x})}\right] , \tag{3}$$

since the average cardinality of the partitioning of X is given by the ratio of the volume of X to that of the mean partition, $2^{H(X)}/2^{H(X|\tilde{X})} = 2^{I(X;\tilde{X})}$, via the standard asymptotic arguments. Notice that this quantity is different from the entropy of the codebook, $H(\tilde{X})$, and this entropy is normally not what we want to minimize.

However, information rate alone is not enough to characterize good quantization since the rate can always be reduced by throwing away details of the original signal x. We need therefore some additional constraints.

2.1 Relevance through distortion: Rate distortion theory

In rate distortion theory such a constraint is provided through a distortion function, $d : X \times \tilde{X} \to \mathcal{R}^+$, which is presumed to be small for good representations. Thus the rate distortion function specifies implicitly what are the most relevant aspects of values in X.

The partitioning of X induced by the mapping $p(\tilde{x}|x)$, has an expected distortion

$$\langle d(x,\tilde{x})\rangle_{p(x,\tilde{x})} = \sum_{x\in X}\sum_{\tilde{x}\in\tilde{X}} p(x,\tilde{x})d(x,\tilde{x}) \ . \tag{4}$$

There is a monotonic tradeoff between the rate of the quantization and the expected distortion: the larger the rate, the smaller is the achievable distortion.

The celebrated rate distortion theorem of Shannon and Kolmogorov (see, for example Ref. [1]) characterizes this tradeoff through the rate distortion function, $R(D)$, defined as the minimal achievable rate under a given constraint on the expected distortion:

$$R(D) \equiv \min_{\{p(\tilde{x}|x):\langle d(x,\tilde{x})\rangle \le D\}} I(X;\tilde{X}) \ . \tag{5}$$

Finding the rate-distortion function is a variational problem that can be solved by introducing a Lagrange multiplier, β, for the constrained expected distortion. One then needs to minimize the functional

$$\mathcal{F}[p(\tilde{x}|x)] = I(X;\tilde{X}) + \beta\langle d(x,\tilde{x})\rangle_{p(x,\tilde{x})} \tag{6}$$

over all normalized distributions $p(\tilde{x}|x)$. This variational formulation has the following well known consequences:

Theorem 1 *The solution of the variational problem,*

$$\frac{\delta\mathcal{F}}{\delta p(\tilde{x}|x)} = 0, \tag{7}$$

for normalized distributions $p(\tilde{x}|x)$, is given by the exponential form

$$p(\tilde{x}|x) = \frac{p(\tilde{x})}{Z(x,\beta)}\exp\left[-\beta d(x,\tilde{x})\right] , \tag{8}$$

where $Z(x,\beta)$ is a normalization (partition) function. Moreover, the Lagrange multiplier β, determined by the value of the expected distortion, D, is positive and satisfies

$$\frac{\delta R}{\delta D} = -\beta \ . \tag{9}$$

Proof. Taking the derivative w.r.t. $p(\tilde{x}|x)$, for given x and $\tilde{x}$, one obtains

$$\frac{\delta\mathcal{F}}{\delta p(\tilde{x}|x)} = p(x)\left[\log\frac{p(\tilde{x}|x)}{p(\tilde{x})} + 1 - \frac{1}{p(\tilde{x})}\sum_{x'} p(x')p(\tilde{x}|x') + \beta d(x,\tilde{x}) + \frac{\lambda(x)}{p(x)}\right] \tag{10}$$

since the marginal distribution satisfies $p(\tilde{x}) = \sum_{x'} p(x')p(\tilde{x}|x')$. $\lambda(x)$ are the normalization Lagrange multipliers for each x. Setting the derivatives to zero and writing $\log Z(x,\beta) = \lambda(x)/p(x)$, we obtain Eq. (8). When varying the normalized $p(\tilde{x}|x)$ the variations $\delta I(X;\tilde{X})$ and $\delta\langle d(x,\tilde{x})\rangle_{p(x,\tilde{x})}$ are linked through

$$\delta\mathcal{F} = \delta I(X;\tilde{X}) + \beta\delta\langle d(x,\tilde{x})\rangle_{p(x,\tilde{x})} = 0 \tag{11}$$

from which Eq. (9) follows. The positivity of β is then a consequence of the concavity of the rate-distortion function (see, for example, Chapter 13 of Ref. [1]). □

2.1.1 The Blahut-Arimoto algorithm

An important practical consequence of the above variational formulation is that it provides a converging iterative algorithm for self-consistent determination of the distributions $p(\tilde{x}|x)$ and $p(\tilde{x})$.

Equations (8) and (1) must be satisfied simultaneously for consistent probability assignment. A natural approach to solve these equations is to alternately iterate between them until reaching convergence. The following lemma, due to Csiszár and Tusnády [5], assures global convergence in this case.

Lemma 2 *Let $p(x,y) = p(x)p(y|x)$ be a given joint distribution. Then the distribution $q(y)$ that minimizes the relative entropy, $D_{KL}[p(x,y)|p(x)q(y)]$, is the marginal distribution $p(y) = \sum_x p(x)p(y|x)$. Namely,*

$$I(X;Y) = D_{KL}[p(x,y)|p(x)p(y)] = \min_{q(y)} D_{KL}[p(x,y)|p(x)q(y)] \ .$$

Equivalently, the distribution $q(y)$ which minimizes the expected relative entropy,

$$\sum_x p(x) D_{KL}[p(y|x)|q(y)],$$

is also the marginal distribution $p(y) = \sum_x p(x)p(y|x)$.

The proof follows directly from the non-negativity of the relative entropy.

This lemma guarantees the marginal condition Eq. (1) through the same variational principle.

Theorem 3 *Equations (1) and (8) are satisfied simultaneously at the minimum of the functional,*

$$\mathcal{F} = -\langle \log Z(x,\beta) \rangle_{p(x)} = I(X;\tilde{X}) + \beta \langle d(x,\tilde{x}) \rangle_{p(x,\tilde{x})} \ , \tag{12}$$

where the minimization is done independently *over the convex sets of the normalized distributions, $\{p(\tilde{x})\}$ and $\{p(\tilde{x}|x)\}$,*

$$\min_{p(\tilde{x})} \min_{p(\tilde{x}|x)} \mathcal{F}\left[p(\tilde{x}); p(\tilde{x}|x)\right] \ .$$

These independent conditions correspond precisely to alternating iterations of Eq. (1) and Eq. (8). Denoting by t the iteration step,

$$\begin{cases} p_{t+1}(\tilde{x}) = \sum_x p(x) p_t(\tilde{x}|x) \\ p_t(\tilde{x}|x) = \frac{p_t(\tilde{x})}{Z_t(x,\beta)} \exp(-\beta d(x,\tilde{x})) \end{cases} \tag{13}$$

where the normalization function $Z_t(x,\beta)$ is evaluated for every t in Eq. (13). Furthermore, these iterations converge to a unique minimum of $\mathcal{F}$ in the convex sets of the two distributions.

For the proof, see references [1, 6]. This alternating iteration is the well known Blauht-Arimoto (BA) algorithm for calculation of the rate distortion function.

It is important to notice that the BA algorithm deals only with the optimal partitioning of the set X given the set of representatives $\tilde{X}$, and not with an optimal choice of the representation $\tilde{X}$. In practice, for finite data, it is also important to find the optimal representatives which minimize the expected distortion, *given* the partitioning. This joint optimization is similar to the EM procedure in statistical estimation and does not in general have a unique solution.

3 Relevance through another variable: The Information Bottleneck

Since the "right" distortion measure is rarely available, the problem of relevant quantization has to be addressed directly, by preserving the *relevant information* about another variable. The relevance variable, denoted here by Y, must not be independent from the original signal X, namely they have positive mutual information $I(X;Y)$. It is assumed here that we have access to the joint distribution $p(x,y)$, which is part of the setup of the problem, similarly to $p(x)$ in the rate distortion case. [1]

3.1 A new variational principle

As before, we would like our relevant quantization $\tilde{X}$ to compress X as much as possible. In contrast to the rate distortion problem, however, we now want this quantization to capture as much of the information about Y as possible. The amount of information about Y in $\tilde{X}$ is given by

$$I(\tilde{X};Y) = \sum_y \sum_{\tilde{x}} p(y,\tilde{x}) \log \frac{p(y,\tilde{x})}{p(y)p(\tilde{x})} \leq I(X;Y). \tag{14}$$

Obviously lossy compression cannot convey more information than the original data. As with rate and distortion, there is a tradeoff between compressing the representation and preserving meaningful information, and there is no single right solution for the tradeoff. The assignment we are looking for is the one that keeps a fixed amount of meaningful information about the relevant signal Y while minimizing the number of bits from the original signal X (maximizing the compression). [2] In effect we pass the information that X provides about Y through a "bottleneck" formed by the compact summaries in $\tilde{X}$.

We can find the optimal assignment by minimizing the functional

$$\mathcal{L}[p(\tilde{x}|x)] = I(\tilde{X};X) - \beta I(\tilde{X};Y), \tag{15}$$

where β is the Lagrange multiplier attached to the constrained meaningful information, while maintaining the normalization of the mapping $p(\tilde{x}|x)$ for every x. At $\beta = 0$ our quantization is the most sketchy possible—everything is assigned to a single point—while as $\beta \to \infty$ we are pushed toward arbitrarily detailed quantization. By varying the (only) parameter β one can explore the tradeoff between the preserved meaningful information and compression at various resolutions. As we show elsewhere [3, 4], for interesting special cases (where there exist sufficient statistics) it is possible to preserve almost all the meaningful information at finite β with a significant compression the original data.

3.2 Self-consistent equations

Unlike the case of rate distortion theory, here the constraint on the meaningful information is *nonlinear* in the desired mapping $p(\tilde{x}|x)$ and this is a much harder variational problem. Perhaps surprisingly, this general problem of extracting the meaningful information—minimizing the functional $\mathcal{L}[p(\tilde{x}|x)]$ in Eq. (15)—can be given an exact formal solution.

Theorem 4 *The optimal assignment, that minimizes Eq. (15), satisfies the equation*

$$p(\tilde{x}|x) = \frac{p(\tilde{x})}{Z(x,\beta)} \exp\left[-\beta \sum_y p(y|x) \log \frac{p(y|x)}{p(y|\tilde{x})}\right], \tag{16}$$

---

[1] The problem of actually obtaining a good enough sample of this distribution is an interesting issue in learning theory, but is beyond the scope of this paper. For a start on this problem see Ref. [4].

[2] It is completely equivalent to maximize the meaningful information for a fixed compression of the original variable.

where the distribution $p(y|\tilde{x})$ in the exponent is given via Bayes' rule and the Markov chain condition $\tilde{X} \leftarrow X \leftarrow Y$, as,

$$p(y|\tilde{x}) = \frac{1}{p(\tilde{x})} \sum_x p(y|x) p(\tilde{x}|x) p(x). \tag{17}$$

This solution has a number of interesting features, but we must emphasize that it is a *formal* solution since $p(y|\tilde{x})$ in the exponential is defined implicitly in terms of the assignment mapping $p(\tilde{x}|x)$. Just as before, the marginal distribution $p(\tilde{x})$ must satisfy the marginal condition Eq. (1) for consistency.

Proof. First we note that the conditional distribution of y on $\tilde{x}$

$$p(y|\tilde{x}) = \sum_{x \in X} p(y|x) p(x|\tilde{x}) \ , \tag{18}$$

follows from the Markov chain condition $Y \leftarrow X \leftarrow \tilde{X}$.[3] The only variational variables in this scheme are the conditional distributions, $p(\tilde{x}|x)$, since other unknown distributions are determined from it through Bayes' rule and consistency. Thus we have

$$p(\tilde{x}) = \sum_x p(\tilde{x}|x) p(x) \ , \tag{19}$$

and

$$p(\tilde{x}|y) = \sum_x p(\tilde{x}|x) p(x|y) \ . \tag{20}$$

The above equations imply the following derivatives w.r.t. $p(\tilde{x}|x)$,

$$\frac{\delta p(\tilde{x})}{\delta p(\tilde{x}|x)} = p(x) \tag{21}$$

and

$$\frac{\delta p(\tilde{x}|y)}{\delta p(\tilde{x}|x)} = p(x|y) \ . \tag{22}$$

Introducing Lagrange multipliers, β for the information constraint and $\lambda(x)$ for the normalization of the conditional distributions $p(\tilde{x}|x)$ at each x, the Lagrangian, Eq. (15), becomes

$$\begin{aligned} \mathcal{L} &= I(X, \tilde{X}) - \beta I(\tilde{X}, Y) - \sum_{x,\tilde{x}} \lambda(x) p(\tilde{x}|x) \qquad (23) \\ &= \sum_{x,\tilde{x}} p(\tilde{x}|x) p(x) \log \left[\frac{p(\tilde{x}|x)}{p(\tilde{x})} \right] - \beta \sum_{\tilde{x},y} p(\tilde{x}, y) \log \left[\frac{p(\tilde{x}|y)}{p(\tilde{x})} \right] - \sum_{x,\tilde{x}} \lambda(x) p(\tilde{x}|x) \ . \end{aligned}$$

Taking derivatives with respect to $p(\tilde{x}|x)$ for given x and $\tilde{x}$, one obtains

$$\begin{aligned} \frac{\delta \mathcal{L}}{\delta p(\tilde{x}|x)} &= p(x) \left[1 + \log p(\tilde{x}|x) \right] - \frac{\delta p(\tilde{x})}{\delta p(\tilde{x}|x)} \left[1 + \log p(\tilde{x}) \right] \qquad (24) \\ &\quad - \beta \left\{ \sum_y \frac{\delta p(\tilde{x}|y)}{\delta p(\tilde{x}|x)} p(y) \left[1 + \log p(\tilde{x}|y) \right] - \frac{\delta p(\tilde{x})}{\delta p(\tilde{x}|x)} \left[1 + \log p(\tilde{x}) \right] \right\} - \lambda(x) \ . \end{aligned}$$

---

[3] It is important to notice that this not a modeling assumption and the quantization $\tilde{X}$ is *not* used as a hidden variable in a model of the data. In the latter, the Markov condition would have been different: $Y \leftarrow \tilde{X} \leftarrow X$.

Substituting the derivatives, Eq.(21) and Eq.(22), and rearranging,

$$\frac{\delta \mathcal{L}}{\delta p(\tilde{x}|x)} = p(x)\left\{\log\left[\frac{p(\tilde{x}|x)}{p(\tilde{x})}\right] - \beta \sum_y p(y|x)\log\left[\frac{p(y|\tilde{x})}{p(y)}\right] - \frac{\lambda(x)}{p(x)}\right\} . \tag{25}$$

Notice that $\sum_y p(y|x)\log\frac{p(y|x)}{p(y)} = I(x,Y)$ is a function of x only (independent of $\tilde{x}$) and thus can be absorbed into the multiplier $\lambda(x)$. Introducing

$$\tilde{\lambda}(x) = \frac{\lambda(x)}{p(x)} - \beta \sum_y p(y|x)\log\left[\frac{p(y|x)}{p(y)}\right] ,$$

we finally obtain the variational condition:

$$\frac{\delta \mathcal{L}}{\delta p(\tilde{x}|x)} = p(x)\left[\log\frac{p(\tilde{x}|x)}{p(\tilde{x})} + \beta \sum_y p(y|x)\log\frac{p(y|x)}{p(y|\tilde{x})} - \tilde{\lambda}(x)\right] = 0 ,$$

which is equivalent to equation (16) for $p(\tilde{x}|x)$,

$$p(\tilde{x}|x) = \frac{p(\tilde{x})}{Z(x,\beta)} \exp\left(-\beta D_{KL}\left[p(y|x)|p(y|\tilde{x})\right]\right) , \tag{26}$$

with

$$Z(x,\beta) = \exp[\beta\tilde{\lambda}(x)] = \sum_{\tilde{x}} p(\tilde{x})\exp\left(-\beta D_{KL}\left[p(y|x)|p(y|\tilde{x})\right]\right) ,$$

the normalization (partition) function. ☐

Comments:

1. Notice that the Kullback-Leibler divergence, $D_{KL}[p(y|x)|p(y|\tilde{x})]$, *emerged* as the relevant "effective distortion measure" from our variational principle but is not assumed otherwise anywhere! It is therefore natural to consider it as the "correct" distortion $d(x,\tilde{x}) = D_{KL}[p(y|x)|p(y|\tilde{x})]$ for quantization in the information bottleneck setting.

2. Equation (26), together with equations (18) and (19), determine self consistently the desired conditional distributions $p(\tilde{x}|x)$ and $p(\tilde{x})$. The crucial quantization is here performed through the conditional distributions $p(y|\tilde{x})$, and the self-consistent equations include also the optimization over the representatives, in contrast to rate-distortion theory, where the selection of representatives is a separate problem.

3.3 The information bottleneck iterative algorithm

As for the BA algorithm, the self consistent equations (16) and (17) suggest a natural method for finding the unknown distributions, at every value of β. Indeed, these equations can be turned into converging, alternating iterations among the three convex distribution sets, $\{p(\tilde{x}|x)\}$, $\{p(\tilde{x})\}$, and $\{p(y|\tilde{x})\}$, as stated in the following theorem.

Theorem 5 *The self consistent equations (18), (19), and (26), are satisfied simultaneously at the minima of the functional,*

$$\mathcal{F}\left[p(\tilde{x}|x); p(\tilde{x}); p(y|\tilde{x})\right] = -\langle \log Z(x,\beta)\rangle_{p(x)} = I(X;\tilde{X}) + \beta\langle D_{KL}[p(y|x)|p(y|\tilde{x})]\rangle_{p(x,\tilde{x})} , \tag{27}$$

where the minimization is done independently *over the convex sets of the normalized distributions,* $\{p(\tilde{x})\}$ *and* $\{p(\tilde{x}|x)\}$ *and* $\{p(y|\tilde{x})\}$. *Namely,*

$$\min_{p(y|\tilde{x})} \min_{p(\tilde{x})} \min_{p(\tilde{x}|x)} \mathcal{F}\left[p(\tilde{x}|x); p(\tilde{x}); p(y|\tilde{x})\right] .$$

This minimization is performed by the converging alternating iterations. Denoting by t the iteration step,

$$\begin{cases} p_t(\tilde{x}|x) = \frac{p_t(\tilde{x})}{Z_t(x,\beta)} \exp(-\beta d(x,\tilde{x})) \\ p_{t+1}(\tilde{x}) = \sum_x p(x) p_t(\tilde{x}|x) \\ p_{t+1}(y|\tilde{x}) = \sum_y p(y|x) p_t(x|\tilde{x}) \end{cases} \tag{28}$$

and the normalization (partition function) $Z_t(\beta, \tilde{x})$ *is evaluated for every t in Eq. (28).*

Proof. For lack of space we can only outline the proof. First we show that the equations indeed are satisfied at the minima of the functional $\mathcal{F}$ (known for physicists as the "free energy"). This follows from lemma (2) when applied to $I(X;\tilde{X})$ with the convex sets of $p(\tilde{x})$ and $p(\tilde{x}|x)$, as for the BA algorithm. Then the second part of the lemma is applied to $\langle D_{KL}[p(y|x)|p(y|\tilde{x})]\rangle_{p(x,\tilde{x})}$ which is an expected relative entropy. Equation (26) minimizes the expected relative entropy w.r.t. to variations in the convex set of the normalized $\{p(y|\tilde{x})\}$. Denoting by $d(x,\tilde{x}) = D_{KL}[p(y|x)|p(y|\tilde{x})]$ and by $\lambda(\tilde{x})$ the normalization Lagrange multipliers, we obtain

$$\delta d(x,\tilde{x}) = \delta\left(-\sum_y p(y|x) \log p(y|\tilde{x}) + \lambda(\tilde{x})(\sum_y p(y|\tilde{x}) - 1) \right) \tag{29}$$

$$= \sum_y \left(-\frac{p(y|x)}{p(y|\tilde{x})} + \lambda(\tilde{x}) \right) \delta p(y|\tilde{x}) . \tag{30}$$

The expected relative entropy becomes,

$$\sum_x \sum_y \left(-\frac{p(y|x)p(x|\tilde{x})}{p(y|\tilde{x})} + \lambda(\tilde{x}) \right) \delta p(y|\tilde{x}) = 0 , \tag{31}$$

which gives Eq. (26), since $\delta p(y|\tilde{x})$ are independent for each $\tilde{x}$. Equation (26) also have the interpretation of a weighted average of the data conditional distributions that contribute to the representative $\tilde{x}$.

To prove the convergence of the iterations it is enough to verify that each of the iteration steps minimizes the same functional, independently, and that this functional is lower-bounded as a sum of two non-negative terms. The only point to notice is that when $p(y|\tilde{x})$ is fixed we are back to the rate-distortion case with fixed distortion matrix $d(x,\tilde{x})$. The argument in [5] for the BA algorithm applies here as well. On the other hand we have just showed that the third equation minimizes the expected relative entropy without affecting the mutual information $I(X;\tilde{X})$. This proves the convergence of the alternating iterations. However, the situation here is similar to the EM algorithm and the functional $\mathcal{F}\left[p(\tilde{x}|x); p(\tilde{x}); p(y|\tilde{x})\right]$ is convex in each of the distribution independently but *not* in the product space of these distributions. Thus our convergence proof does not imply uniqueness of the solution. □

3.4 The structure of the solutions

The formal solution of the self consistent equations, described above, still requires a specification of the structure and cardinality of $\tilde{X}$, as in rate distortion theory. For every value of the Lagrange multiplier β there are corresponding values of the mutual information $I_X \equiv I(X, \tilde{X})$, and $I_Y \equiv I(\tilde{X}, Y)$ for every choice of the cardinality of $\tilde{X}$. The variational principle implies that

$$\frac{\delta I(\tilde{X}, Y)}{\delta I(X, \tilde{X})} = \beta^{-1} > 0 \ , \tag{32}$$

which suggests a *deterministic annealing* approach. By increasing the value of β one can move along *convex* curves in the "information plane" (I_X, I_Y). These curves, analogous to the rate-distortion curves, exists for every choice of the cardinality of $\tilde{X}$. The solutions of the self-consistent equations thus correspond to a family of such annealing curves, all start from the (trivial) point $(0, 0)$ in the information plane with infinite slope and are parameterized by the value of β. Interestingly, every two curves in this family separate (bifurcate) at some finite (critical) β through a second, through a second-order phase transition. These transitions form an hierarchy of relevant quantizations for different cardinalities of $\tilde{X}$, as described in [2, 3, 4].

Further work

The most fascinating aspect of the information bottleneck principle is that it provides a unified framework for different information processing problems, including prediction, filtering and learning [4]. There are already several successful applications of this method to various "real" problems, such as semantic clustering of English words [2], document classification [3], neural coding, and spectral analysis.

Acknowledgements

Helpful discussions and insights on rate distortion theory with Joachim Buhmann and Shai Fine are greatly appreciated.

References

[1] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, New York: Wiley, 1991.

[2] F. C. Pereira, N. Tishby, and L. Lee, "Distributional clustering of English words", in *30th Annual Mtg. of the Association for Computational Linguistics*, 183–190, 1993.

[3] N. Slonim and N. Tishby, "Agglomerative Information Bottleneck", To appear in Proc. of *Neural Information Processing systems (NIPS-12)* 1999.

[4] W. Bialek and N. Tishby, "Extracting Relevant Information", In preparation.

[5] I. Csiszár and G. Tusnády, "Information geometry and alternating minimization procedures"*Statistics and Decisions*, Supplement Issue 1: 205–237, 1984.

[6] R. E. Blahut, "Computation of Channel capacity and rate distortion function", *IEEE Trans. Inform. Theory,*, IT-18:460–473, 1972.

Theory Meets Practice: Universal Source Coding with the Burrows Wheeler Transform*

Michelle Effros

Dept. of Electrical Engineering
California Institute of Technology
Pasadena, CA 91125
effros@caltech.edu

Abstract

Recent work in source coding with the Burrows Wheeler Transform (BWT) demonstrates the asymptotic optimality of a collection of new BWT-based lossless source codes when applied to stationary ergodic sources. The resulting rate of convergence bounds fall off more quickly than the known bounds for Ziv-Lempel codes (LZ'77, LZ'78, and their descendants). This work treats the practical coding performance of two of these new algorithms and explores other applications for the BWT suggested by these algorithms. Results include experimental demonstration of the theoretically derived $O(\log n/n)$ and $O(\sqrt{\log n/n})$ rates of convergence for the new BWT-based source codes; analyses and demonstrations of the potential shortcomings of these algorithms for some practical source coding applications; and a brief description of the causes of these discrepancies. Finally, the new BWT-based codes suggest extremely simple techniques for answering questions that might otherwise be considered intractable due to their reliance on higher order statistical models. A short discussion, with examples, is included.

1 Introduction

In the five years since its introduction, the BWT [1] has received an enormous amount of attention in the practical lossless source coding literature. This popularity results from the low computational complexity and good coding performance of BWT-based codes. In an attempt to gain a better understanding of how and how well BWT-based codes work, [2] and [3] present a theoretical analysis of BWT outputs and the resulting implications for existing and new BWT-based codes. An existing BWT-based source code is proven universal when applied to extensions on the source alphabet. The resulting code achieves $O(\log\log n/\log n)$ convergence in coding sequences of length n from an arbitrary finite memory source. The new algorithms introduced in [2, 3] require no alphabet extensions and are also universal, with rates of convergence ranging from $O(\sqrt{\log n/n})$ to $O(\log n/n)$. These performance bounds surpass the performance bounds for the earlier algorithm and the $O(\log\log n/\log n)$ and $O(1/\log n)$ performance bounds for Ziv-Lempel

---

*This research is supported in part by NSF CAREER Grant No. MIP-9501977, a grant from the Powell Foundation, and donations through the Intel 2000 Technology for Education Program

	Step 1							Step 2							Step 3
1	**a**	**n**	**a**	**n**	**a**	**b**	**§**	a	b	§	a	n	a	n	n
2	§	a	n	a	n	a	b	a	n	a	b	§	a	n	n
3	b	§	a	n	a	n	a	**a**	**n**	**a**	**n**	**a**	**b**	**§**	
4	a	b	§	a	n	a	n	b	§	a	n	a	n	a	a
5	n	a	b	§	a	n	a	n	a	b	§	a	n	a	a
6	a	n	a	b	§	a	n	n	a	n	a	b	§	a	a
7	n	a	n	a	b	§	a	§	a	n	a	n	a	b	b

Figure 1: The BWT of the sequence "banana". The unshifted data sequence "ananab§" appears in boldface type. Since that sequence occupies row 3 of the lexicographically ordered table (Step 2) $BWT_{\mathbb{N}}(\text{banana}) = 3$. The final column of the table – excluding row $BWT_{\mathbb{N}}(\text{banana}) = 3$ – contains the sequence "nnaaab". Hence $BWT_{\mathcal{X}}(\text{banana}) = \text{nnaaab}$.

codes [4, 5, 6]. The faster convergence comes within a constant factor of the optimal rate of convergence for the given class of sources.

The source coding algorithms described above demonstrate just one use of the BWT as a tool for managing the explosive growth in complexity associated with using distribution models with growing memory. This observation leads to very low complexity and low memory algorithms for an enormous array of applications. This work includes both an empirical performance analysis of the newly introduced BWT-based source codes and an introduction of the BWT as a tool for use outside of source coding. Section 2 contains a brief description of the BWT. Section 3 gives background on the statistical properties of the BWT output and the proposed algorithms that use these properties for source coding. Alternative applications for the BWT and experimental analyses of the source coding algorithms appear in Section 4. A summary of conclusions follows in Section 5.

2 The Burrows Wheeler Transform

The Burrows Wheeler Transform [1] is a reversible block-sorting transform that operates on a sequence of n data symbols to produce a permuted data sequence of the same symbols and a single integer in $\{1, \ldots, n+1\}$. Let $BWT_n : \mathcal{X}^n \to \mathcal{X}^n \times \{1, \ldots, n+1\}$ denote the n-dimensional BWT function and $BWT_n^{(-1)} : \mathcal{X}^n \times \{1, \ldots, n+1\} \to \mathcal{X}^n$ denote the inverse of BWT_n. Since the sequence length n is evident from the source argument, the functional transcript is typically dropped. The notations $BWT_{\mathcal{X}}$ and $BWT_{\mathbb{N}}$ denote the character and integer portions of the BWT, respectively. An implementation of the BWT with $O(n)$ worst-case complexity and memory appears in [1].

For the purposes of this work, the forward BWT operation begins with a time-reversal of data sequence X^n and the concatenation of an end-of-file symbol $\S \notin \mathcal{X}$. The forward transform then proceeds by forming all $(n+1)$ cyclic shifts of the altered data string and sorting those cyclic shifts lexicographically. (The lexicographic order of symbol § in $\mathcal{X} \cup \{\S\}$ may be chosen arbitrarily. In this work, that symbol is ordered last.) The index $u = BWT_{\mathbb{N}}(x^n)$ denotes the location of the unshifted data string in the lexicographically ordered list. The output $y^n = BWT_{\mathcal{X}}(x^n)$ gives the last character of all but the uth (lexicographically ordered) cyclic shift. Since the last character of the uth shift must equal §, that character need not be described. An example of the BWT of the word

	Step						
	1	2	3	4	5	6	7
1	······ n	a ····· n	ab ···· n	ab§ ··· n	ab§a ·· n	ab§an · n	ab§anan
2	······ n	a ····· n	an ···· n	ana ··· n	anab ·· n	anab§ · n	anab§an
3	······ §	a ····· §	an ···· §	ana ··· §	anan ·· §	anana · §	**ananab§**
4	······ a	b ····· a	b§ ···· a	b§a ··· a	b§an ·· a	b§ana · a	b§anana
5	······ a	n ····· a	na ···· a	nab ··· a	nab§ ·· a	nab§a · a	nab§ana
6	······ a	n ····· a	na ···· a	nan ··· a	nana ·· a	nanab · a	nanab§a
7	······ b	§ ····· b	§a ···· b	§an ··· b	§ana ·· b	§anan · b	§ananab

Figure 2: The inverse BWT for the pair (nnaaab,3) from Figure 1. In this example, $n = |\text{banana}| = 7$ and row 3 (shown in bold in step 18) is specified. To find the output of the inverse BWT, delete the § from row 3 of the final table and then reverse the remaining string. $BWT^{-1}(\text{nnaaab}, 3) = \text{banana}$.

"banana" appears in Figure 1. Here $BWT(\text{banana}) = (\text{nnaaab}, 3)$. Note that despite the use of the end-of-file symbol $\S \notin \mathcal{X}$, $BWT_{\mathcal{X}}(x^n) \in \mathcal{X}^n$ for all $x^n \in \mathcal{X}^n$.

For the BWT to be a *reversible* sequence transformation, it must be possible to reconstruct the full table of lexicographically ordered cyclic shifts using only the combined information from $BWT_{\mathcal{X}}(x^n)$ and $BWT_{\mathbb{N}}(x^n)$. An example of the inverse BWT of the pair (nnaaab,3) from the example in Figure 1 appears in Figure 2. Intuitively, this reconstruction proceeds column by column as follows. Constructing the *last* column of the table requires only the insertion of the symbol § between the $(u-1)$th and uth characters in data string $BWT_{\mathcal{X}}(x^n)$ (Step 1). By the table construction, the first column of the table is simply an ordered copy of the last column of the table. Thus given the table's last column, the *first* column reconstruction requires only an alphabetization of the list found in the last column (Step 2). To reconstruct the second column, notice that each row is a cyclic shift of every other row, and hence that the last and first columns together provide a list of all consecutive pairs of letters. Lexicographically ordering this list of consecutive pairs yields the (first and) *second* column(s) of the table (Step 3). Repeating the above process for all triples, quadruples, etc. sequentially reproduces all columns of the original data matrix. Deleting the § from the end of row u and reversing the remaining string gives the original data sequence.

3 Background

BWT Properties

Let $(Y^n, U) = BWT(X^n)$. If X^n gives samples drawn from a finite-memory source, then the output Y^n looks like a piecewise independent and identically distributed (p.i.i.d.) data sequence. The number of i.i.d. "pieces" varies with the source distribution, but for a fixed source does not vary with the length n of the data sequence X^n [2, 3]. The p.i.i.d. property may be understood intuitively by noting that in finite memory sources, the distribution on each symbol depends only on some finite collection of past symbols, called a "context." In alphabetizing the reversed cyclic shifts of a given data sequence, the BWT table groups together all rows that begin with the same reversed context. Thus all symbols following that context in the original data string (or all symbols preceding that context in the reversed data string) appear consecutively in the BWT output Y^n.[1]

Source Coding Algorithms

Building on the p.i.i.d. property discussed above, [2] and [3] describe a collection of new lossless BWT-based codes. Descriptions of two of the conceptually and computationally simplest algorithms follow. Both of the algorithms are $O(n)$ in computation and memory and universal for finite-memory sources and stationary ergodic sources on finite alphabets [2, 3]. The algorithms use independent codes for describing U and Y^n. Assuming that the decoder knows the sequence length n, the natural $(\lfloor \log(n+1) \rfloor + 1)$-bit binary expansion suffices for describing U. The algorithms differ in their methods for describing Y^n. Since both algorithms code Y^n rather than coding X^n directly, neither of the algorithms is a sequential code. However, in both algorithms, the sequence length n may be chosen based on constraints on computation, memory, or delay, allowing the n-vectors that compose the original data sequence to be encoded and decoded sequentially.

Redundancy bounds are used in comparing the performances of these algorithms to each other and to those of other universal algorithms. The redundancy term $\Delta_n(\theta)$ denotes the difference between the expected rate per symbol and the optimal performance when coding samples from P_θ. Thus if $\ell(X^n)$ is the length of a file used to losslessly describe X^n, $\Delta_n(\theta) = (1/n)E_\theta \ell(X^n) - H_\theta(\mathcal{X})$. Given a stationary finite-memory source P_θ with state-space $\mathcal{S}$ and source alphabet $\mathcal{X}$, the optimal redundancy equals $(|\mathcal{X}| - 1)|\mathcal{S}| \log n/(2n) + O(1/n)$ [8, 9].

A Known State-Space Algorithm

The simplest algorithm considered in [2, 3] relies on an estimate $\hat{\mathcal{S}}$ of the finite-memory source's state-space $\mathcal{S}$. The choice $\hat{\mathcal{S}} = \mathcal{X}^{\hat{M}}$ requires only an estimate $\hat{M}$ of the source's memory constraint M. Allowing $\hat{M}$ to grow slowly with the sequence length n makes the code universal for both stationary finite-memory sources and more general stationary ergodic sources [3]. A discussion of the choice of the growth rate of $\hat{M}$ appears in [3].

We call an estimate $\hat{\mathcal{S}}$ (or sequence of estimates $\{\hat{\mathcal{S}}_n\}$) of true state-space $\mathcal{S}$ "accurate" when for each $s \in \mathcal{S}$ there exists an $\hat{s} \in \hat{\mathcal{S}}$ (or $\hat{s} \in \hat{\mathcal{S}}_n$ for large enough n) such that s is a suffix of $\hat{s}$. (By this requirement, modeling a Markov-2 source as Markov-3 is considered an accurate model, but modeling a Markov-2 source as Markov-1 is not.) When the model is inaccurate, the code is not universal, but its performance approaches the best possible performance achievable with the given model. (Thus in modeling a Markov-2 source as Markov-1, the coding performance approaches $H(X_2|X_1)$ rather than $H(\mathcal{X}) = H(X_3|X^2)$.) Let $\hat{\mathcal{S}}' = \{\hat{s}_1, \hat{s}_2, \ldots, \hat{s}_C\}$ denote the contexts appearing in data string X^n, and assume that the contexts are ordered such that for any $1 \leq j \leq C-1$, the time-reversal of $\hat{s}_j$ precedes the time-reversal of $\hat{s}_{j+1}$ lexicographically. Note that $\hat{\mathcal{S}}'$ may be a smaller set than set $\hat{\mathcal{S}}$ (if some states never appear in X^n) or a slightly larger set than $\hat{\mathcal{S}}$ (since the contexts of the first few symbols of X^n may be truncated by the beginning of the data string – leaving a modified context beginning with the boundary symbol §).

By construction of the BWT encoding table (see, for example, Figure 1), for any $1 \leq j \leq C-1$ all appearances of the time-reversal of $\hat{s}_j$ in columns 1 through $|\hat{s}_j|$ of the encoding table must precede all appearances of the time-reversal of $\hat{s}_{j+1}$ in columns 1 through $|\hat{s}_{j+1}|$. For each $1 \leq j \leq C$, let n_j equal the number of appearances of the time-reversal of $\hat{s}_j$ in columns one through $|\hat{s}_j|$ of the table. Exclude row U from this calculation, so that $\sum_{j=1}^{C} n_j = n$. Note that $n_j > 1$ for at most $|\hat{\mathcal{S}}|$ values of n_j.

---

[1] The time-reversal is actually unnecessary to preserve this property, since the time-reversal of any finite-memory source is also a finite memory source [7]. A longer discussion on this point appears in [3]. The time-reversal convention is included here for convenience.

Algorithm

1. For each $1 \leq j \leq C$, send "00" if $n_j > 1$, and "11" otherwise.
2. Send the string "01" (to denote that all subsequences have been described).
3. For all but the last j such that $n_j > 1$, use $\lceil \log n \rceil$ bits to specify n_j.
4. For each $1 \leq j \leq C$, describe the next length-n_j subsequence independently of all other subsequences using an adaptive arithmetic code [10] with the Krichevsky-Trofimov [11] probability model (or any other universal code with optimal convergence on i.i.d. data sequences).

The Krichevsky-Trofimov estimate uses $|\mathcal{X}|$ counters $\{r(x) : x \in \mathcal{X}\}$. Let $r_i(x)$ denote the value of counter $r(x)$ after seeing the ith symbol x_i in $x^n = x_1, x_2, \ldots, x_n$. Set $r_0(x) = 1/2$ for each $x \in \mathcal{X}$. Then at each time $i \geq 1$, increment the counter corresponding to symbol x_i, leaving the remaining counters unchanged. Thus for any $i \geq 0$, $r_i(x) = (1/2) + \sum_{k=1}^{i} 1(x_k = x)$, where $1(\cdot)$ denotes the indicator function. The Krichevsky-Trofimov probability estimate takes the form $P_c(x^n) = \prod_{k=1}^{n}(r_{k-1}(x_k)/(\sum_{x\in\mathcal{X}} r_{k-1}(x)))$.

Theorem 1 **([2, 3])** *Given a finite-memory source with alphabet $\mathcal{X}$ and state-space $\mathcal{S}$, if $S = \min\{|\hat{\mathcal{S}}|, |\hat{\mathcal{S}}'|\}$, then the known state-space algorithm achieves redundancy bounded as*

$$\Delta_n(\theta) \leq \frac{S(|\mathcal{X}|+1)}{2}\frac{\log n}{n} + O\left(\frac{1}{n}\right)$$

bits per symbol for all θ in class of finite-memory sources when the estimate $\hat{\mathcal{S}}$ is accurate. When $\hat{\mathcal{S}}$ is not accurate, the performance approaches the optimal performance achievable for that model *with the rate of convergence given above.*

The rate of convergence described in Theorem 1 differs from Rissanen's optimal rate of convergence by a factor of $(S(|\mathcal{X}|+1))/(|\mathcal{S}|(|\mathcal{X}|-1))$. When $\hat{\mathcal{S}} = \mathcal{S}$, this factor takes on values in $(1, 1.5]$, giving $129/127 = 1.02$ for text compression using the ASCII alphabet ($|\mathcal{X}| = 128$). When $|\hat{\mathcal{S}}| > |\mathcal{S}|$, the factor can be arbitrarily large.

A Finite Memory Code

Recall that the number of subsequences in the p.i.i.d. data does not grow with n. In particular, for large n, the length of the subsequence associated with prefix s should approximate $p(s)n$ by the law of large numbers. Thus, for large n, the data sequence Y^n breaks down roughly into $|S|$ subsequences, each of a length proportional to n. Even without the knowledge of the boundary points between these subsequences, if we break the data string into small enough pieces, most of these pieces should contain samples from only one conditional distribution, and thus each such piece can be easily compressed. Let $w(n)$ denote the length of these "pieces." Then the algorithm proceeds as follows.

Algorithm

1. Break the data sequence into consecutive subsequences of length $w(n)$.
2. For each $1 \leq j \leq \lfloor n/w(n) \rfloor$, describe the next length-$w(n)$ subsequence independently of all other subsequences using an adaptive arithmetic code with

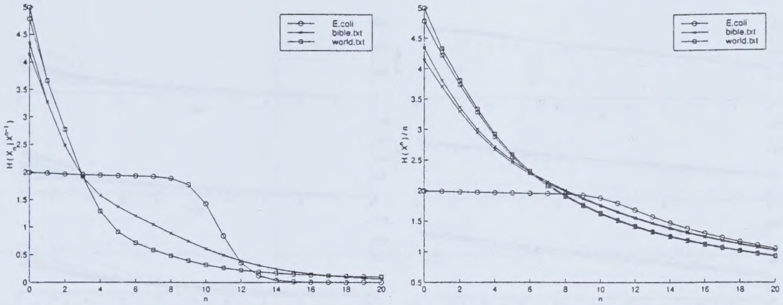


Figure 3: Conditional entropy $H(X_n|X^{n-1})$ (left) and per symbol entropy $H(X^n)/n$ (right) as a function n on the empirical distributions in three files from the Canterbury large file corpus: E.coli (o), bible.txt (x), and world192.txt (box). Entropies calculated using both the forward and the reverse data string are shown. The files contain 4.63, 4.04, and 2.47 million characters, respectively.

the Krichevsky-Trofimov probability model (or another such suitable code).

3. Describe any remaining subsequence independently of all other subsequences using an adaptive arithmetic code with the Krichevsky-Trofimov probability model (or another such suitable code).

Theorem 2 ([2, 3]) *Given a finite-memory source with unknown state-space $\mathcal{S}$ and unknown memory constraint M, the above algorithm achieves redundancy bounded as*

$$\Delta_n(\theta) \leq \sqrt{(|\mathcal{S}|-1)(|\mathcal{X}|-1)\log|\mathcal{X}|}\sqrt{\frac{\log n}{n}} + O\left(\frac{\log\log n}{\sqrt{n \log n}}\right)$$

with $w(n) = (1/2)\sqrt{(|\mathcal{X}|-1)\log|\mathcal{X}|/(|\mathcal{S}|-1)}\sqrt{n \log n}$, which depends on $|\mathcal{S}|$. When the memory length cannot depend on $|\mathcal{S}|$, $w(n) = \sqrt{n \log n}$ gives $\Delta_n(\theta) \leq O(\sqrt{\log n/n})$.

The analysis leading to Theorem 2 does not take into account several pleasing characteristics of the described algorithm. In particular, for many sources, such as text, neighboring distributions in the BWT output often have very similar statistics due to their similar contexts. Thus even in regions overlapping more than one distribution, the performance of the above code should be quite good.

4 Ramifications – Theory and Practice

BWT Properties

The properties described in Section 3 make the BWT a powerful tool for a wide variety of applications. For example, the BWT makes feasible analyses on high dimensional distributions that would be prohibitive by more conventional means. To illustrate this point, consider calculating the empirical conditional entropy $H(X_n|X^{n-1})$ of a file for larger and larger values of n. Direct calculation of this value is difficult for large n due to the need to simultaneously calculate $|\mathcal{X}|^{n-1}$ distinct conditional probabilities (corresponding

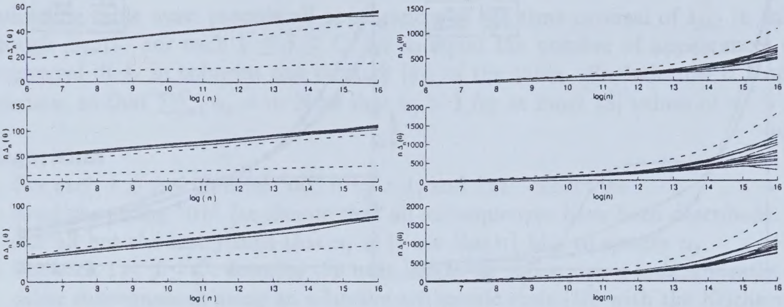


Figure 4: Total redundancy results using the known state-space algorithm (left) and finite memory algorithm (right). From top to bottom: Results on Markov-1 with $\mathcal{X} = \{0, 1\}$, Markov-2 with $\mathcal{X} = \{0, 1\}$, and Markov-1 with $\mathcal{X} = \{0, 1, 2\}$ sources. The solid lines give the experimental performance on all 16 sources of each class; the dash-dot lines indicate the first-order terms predicted by Theorems 1, Theorem 2.

to all $x^{n-1} \in |\mathcal{X}|^{n-1}$). The memory required for such a task grows exponentially in n. In contrast, by grouping together all instances of context x^{n-1}, the BWT allows calculation of $H(X_n|X^{n-1})$ by tracking only one conditional distribution at a time. Thus calculation of the entropy $H(X_{20}|X^{19})$ becomes no more difficult than calculation of $H(X_2|X_1)$, despite the $|\mathcal{X}|^{19}$ distributions – each of $|\mathcal{X}|$ symbols – involved in this calculation. Figure 3 shows $H(X_n|X^{n-1})$ and $H(X^n)/n$ for $n \in \{0, \ldots, 20\}$ ($n = 0$ corresponds to the case without memory) for three files from the Canterbury large file data set. Note that the conditional entropy goes to zero for large n because the (2-4 Mbyte) files provide too few samples from each conditional distribution for large values of n.

Algorithmic Performance

The previous results summarized in the theorems of Section 3 bound the performance of two new BWT-based source codes on data drawn from finite memory sources. They do not, however, provide a direct analysis of BWT-based codes like bzip or prove that the finite memory source model accurately describes natural data sources such as text. This section treats these questions by examining empirical results comparing the performance of these new algorithms to both the theoretical bounds and the performance of a variety of existing algorithms. Results on both real and synthetic data sets are included.

Synthetic Data Sources

Consider coding samples from a variety of synthetic finite-memory sources. Three classes are considered. For each class of data, 65536 samples are drawn from each of 16 independently chosen sources. In the first class, the sources are Markov-1 sources on alphabet $\mathcal{X} = \{0, 1\}$. For each source, the probabilities $p(0|0)$ and $p(0|1)$ are drawn independently and uniformly on $(0, 1)$. The second class contains Markov-2 source on alphabet $\mathcal{X} = \{0, 1\}$. For each source, the probabilities $p(0|00)$, $p(0|01)$, $p(0|10)$, and $p(0|11)$ are drawn independently and uniformly on $(0, 1)$. The third class contains Markov-1 sources on alphabet $\mathcal{X} = \{0, 1, 2\}$. The probability vectors $(p(0|0), p(1|0), p(2|0))$, $(p(0|1), p(1|1), p(2|1))$, and $(p(0|2), p(1|2), p(2|2))$ are drawn independently and uniformly on the probability simplex $\Lambda = \{(p_0, p_1, p_2) : p_0 + p_1 + p_2 = 1, p_0, p_1, p_2 > 0\}$.

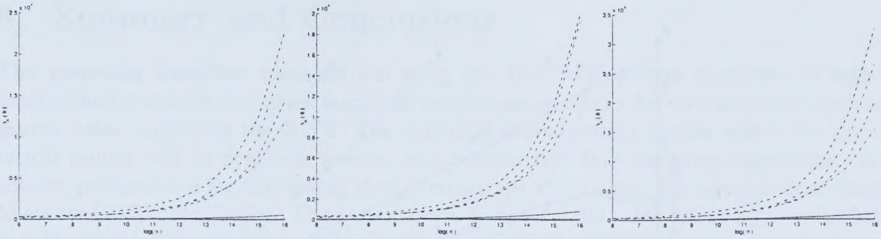


Figure 5: Average total redundancy results. From left to right: Results on Markov-1 with $\mathcal{X} = \{0, 1\}$, Markov-2 with $\mathcal{X} = \{0, 1\}$, and Markov-1 with $\mathcal{X} = \{0, 1, 2\}$ sources. The solid lines give the experimental performance of the algorithms described in this work; the dash-dot lines give the performance of several Ziv-Lempel codes; and the dashed lines give the performance of earlier BWT-based codes.

Figure 4 compares the total rate redundancy $n\Delta(X^n|\theta) = \ell(X^n) - nH_\theta(\mathcal{X})$ for each source with the theoretically predicted value. Here $H_\theta(\mathcal{X})$ is the analytically calculated entropy based on the empirical distribution associated with the n samples from P_θ. In each case, the solid lines give the experimental performance on all 16 sources of each class; the dash-dot lines indicate the *first-order* terms predicted by Theorems 1 and 2; and the dashed lines (included only for the first algorithm) indicate the optimal rate of convergence. The known state-space results use $\hat{\mathcal{S}} = \mathcal{S}$. While in the case of the second algorithm, the predicted bounds seem to be somewhat loose, overall the empirical performances match well with the predicted results. In particular, they support the notion that the relative performances predicted by the bounds hold in practice, with the known state-space algorithm achieving better empirical performance than the finite memory algorithm.

Figure 5 compares the total rate redundancy – averaged over all 16 sources of each class – of the algorithms described in this work to those of other universal codes. In this figure, the solid lines give the experimental performance of the algorithms described in this work; the dash-dot lines give the performance of several Ziv-Lempel codes; and the dashed lines give the performance of earlier BWT-based algorithms. The total redundancies of the proposed algorithm are several orders of magnitude smaller than those of the other algorithms at large values of n.

Natural Data Sources

Experimental results comparing the performances of BWT-based codes with those of Ziv-Lempel style algorithms on the Calgary corpus appear in Figure 6. Two variations of the finite memory code appear in that figure. The first is the algorithm described in Section 3. The second modifies that algorithm only by the use of escape characters. This modification allows shorter descriptions for letters previously seen in the current subsequence in exchange for longer descriptions for symbols as yet unseen (by first describing an "escape" and then describing the symbol). Experimental results for the known state-space algorithm with $\hat{S} = \mathcal{X}^{\hat{M}}$ and $\hat{M} \in \{1, 2, 3\}$ are also included. While the slopes of these curves suggest that they surpass their competitors for large enough n, the performances at the values of n given are poor.

While the known state-space and finite memory algorithms of Section 3 can be made to outperform their competitors using methods similar to those incorporated in other

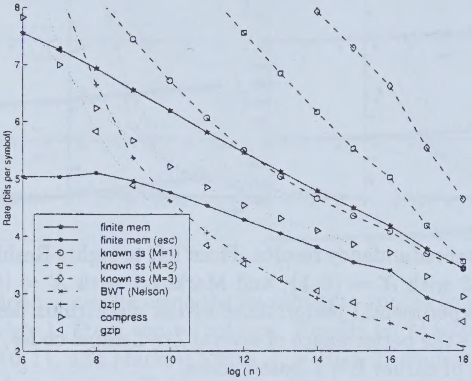


Figure 6: Average rate achieved by treating each file in the Calgary corpus as a sequence of length-n subfiles, and coding each of those subfiles independently. Algorithms tested include two variations on the finite-memory code (labeled "finite mem" and drawn with solid lines), the known state-space algorithm for $\hat{M} \in \{1, 2, 3\}$ (labeled "known ss" and drawn with dashed lines), two existing BWT algorithms (labeled "BWT Nelson" and "bzip" and drawn with dash-dot lines), and two Ziv-Lempel style codes (labeled "compress" and "gzip" and drawn with dotted lines).

BWT-based codes, the cause for the performance results described above deserves attention. The fact that the known state space algorithm and the finite memory code achieve redundancies several orders of magnitude better than those of their competitors on *synthetic sources* while achieving far less impressive performance on *natural sources* results from several key factors. First, note that while the $O(\log n/n)$ redundancy bound of the known state-space algorithm is far faster than the $O(1/\log n)$ bound on the best Ziv-Lempel codes, when the constant in front of $\log n/n$ is very large, the performance may still be bad for the values of n seen in practice. In particular, even at $\hat{M} = 3$ (a low estimate for M, which is commonly believed to be at least 6 for English text), Theorem 1 bounds the convergence by $|\hat{S}|(|\mathcal{X}|+1)/2 \approx 256^3 \times 256/2 = 2^{31}$. Since the file sizes in the Calgary corpus range from around 2^{13} to 2^{20} symbols, this factor is alarming. While this bound is overly pessimistic (the true rate of convergence depends on the number S of patterns *seen* rather than $|\hat{S}|$ – a number nowhere greater than $256^2 = 2^{16}$ in the Calgary corpus and closer to 2^{12} for most files given there) the effect of this factor is nonetheless consequential, as the experimental results indicate.

The final observation arising from these experimental results is that the finite memory source model is in some ways inadequate for modeling some natural source statistics. In particular, while the assumption of finite memory seems reasonable for most text files, the assumption that the source alphabets are identical for all conditional distributions seems a poor model for text, where the collection of characters that can follow a given context are severely limited. While entropy codes *eventually* remove the effects of such symbols by making their probabilities go to zero, the short-term impact can be severe. Many of the methods employed in algorithms like bzip and varieties of PPM focus precisely on this problem. Better source models – perhaps involving conditional distributions on random subsets of the original source alphabet – would likely lead to better algorithms.

5 Summary and Conclusions

The preceding describes methods for using the BWT to analyze functions of high-dimensional statistics and gives empirical performances results for two universal lossless source codes employing the BWT. The empirical source coding results match the theoretical results well on synthetic sources and demonstrate that the given algorithms can achieve performance far surpassing that of either the well-known Ziv-Lempel algorithms or earlier BWT-based codes. The performance on natural data sets fail to achieve the full benefits promised by the asymptotic results, due at least in part to their non-asymptotic nature. A further cause of this discrepancy seems to arise from differences between the characteristics of the natural sources and the assumed model. A brief discussion of a potential variation on the finite memory source model is included.

References

[1] M. Burrows and D. J. Wheeler. A block-sorting lossless data compression algorithm. Technical Report SRC 124, Digital Systems Research Center, Palo Alto, CA, May 1994.

[2] M. Effros. Universal lossless source coding with the Burrows Wheeler transform. In *Proceedings of the Data Compression Conference*, pages 178–187, Snowbird, UT, March 1999. IEEE.

[3] M. Effros. Universal Burrows-Wheeler source coding. 1999. Submitted to the *IEEE Transactions on Information Theroy* June 28, 1999.

[4] A. D. Wyner and A. J. Wyner. Improved redundancy of a version of the Lempel-Ziv algorithm. *IEEE Transactions on Information Theory*, IT-41(3):723–732, May 1995.

[5] G. Louchard and W. Szpankowski. On the average redundancy rate of the Lempel-Ziv code. *IEEE Transactions on Information Theory*, IT-43(1):1–8, January 1997.

[6] S. A. Savari. Redundancy of the Lempel-Ziv incremental parsing rule. *IEEE Transactions on Information Theory*, IT-43(1):9–21, January 1997.

[7] G. Seroussi and M. Weinberger. On tree sources, finite state machines, and time reversal. In *Proceedings of the IEEE International Symposium on Information Theory*, page 390, Whistler, British Columbia, September 1995.

[8] J. Rissanen. Universal coding, information, prediction, and estimation. *IEEE Transactions on Information Theory*, 30(4):629–636, July 1984.

[9] J. Rissanen. Complexity of strings in the class of Markov processes. *IEEE Transactions on Information Theory*, IT-32(4):526–532, July 1986.

[10] F. Jelinek. Buffer overflow in variable length coding of fixed rate sources. *IEEE Transactions on Information Theory*, 14:490–501, May 1968.

[11] R.E. Krichevsky and V.K. Trofimov. The performance of universal encoding. *IEEE Transactions on Information Theory*, IT-27(2):199–207, 1981.

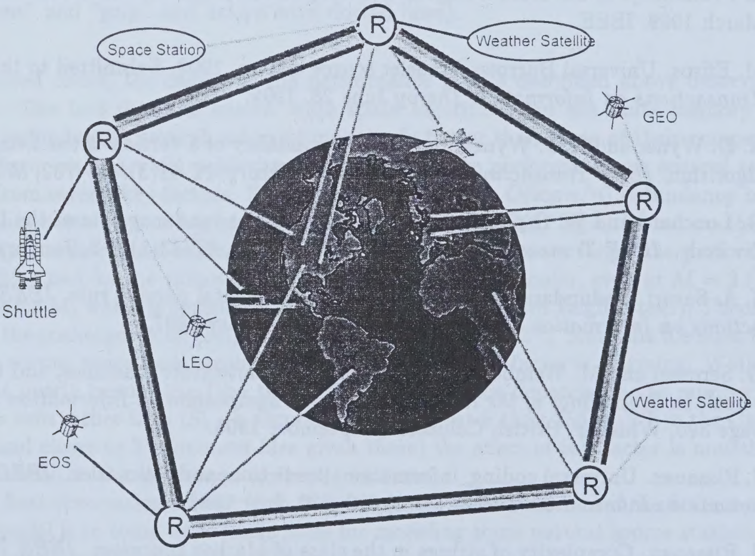
Optical Space Communications and Networking

(Invited paper)

Vincent W. S. Chan
Laboratory for Information and Decision Systems
Department of Aeronautics & Astronautics
and
Department of Electrical Engineering & Computer Science
MIT
Room 35-304, 77 Mass. Ave,
Cambridge, Mass. 02139
Phone: 617-258-8222, email: chan@MIT.edu

Abstract

Optical space communication technology has been under development for over thirty years. The technology promises high capacity, low power consumption, light weight, small sizes and low costs for satellite crosslinks. It will be a key and critical building block for wide area space networks, supporting mobile users (terrestrial, airborne and spaceborne), high speed data services for small satellite terminals and serving as a backbone network for high speed trunking as depicted pictorially in the following figure.



The system of satellites can be in geosynchronous orbits (GEOs), medium earth orbits (MEOs) and low earth orbits (LEOs), interconnected via optical crosslinks and providing radio frequency access links. Data rates of the order of 10 Gb/s or higher are required in this application and microwave crosslinks are too large and costly to be practical at these rates. If successfully developed and deployed, this type of space networks will provide unique and attractive attributes and also can become an economical alternative to long-haul fiber transmissions, especially for intercontinental connections. However, in spite of over a billion dollars in accumulated research and development over thirty years, the technology has not been deemed ready until recently. The reason is that there are three areas of critical technologies and system architectures that had to be developed before the system is viable. These areas are:

1. *Spatial acquisition and tracking of very narrow (~10μr) optical beams:*

Initial pointing uncertainties are of the order of 1 – 5 mr and dynamic jitter is between 1 – 100 μr, depending on the spacecraft design. Since communications beam widths need to be of the order of 5 – 20 μr, a cooperative acquisition subsytem would pull the communication beam into the locking range of an active feedback control system with the aid of a beacon from the receiving platform. The residual errors need to be of the order of 1/20 of a beamwidth for robust communication performance.

2. *Design of a robust and stable optical/mechanical/thermal subsystem*

A sophisticated optical/mechanical/thermal subsystem is necessary to maintain diffraction-limited performance. This design will have to survive tough launch loads and operate on-orbit in widely varying thermal conditions and in the presence of dynamic jitter.

3. *Design of a robust, light weight and high speed optical communications subsystem*

Recent adaptation of 1.5 μm telecommunications optical communication hardware has helped tremendously the maturing process of the space technology. However, there are two additional aspects that had to be developed for space applications. It is due to the fact that, as opposed to terrestrial systems, putting an amplifier in midstream is not practical. Thus the transmitter needs a high power optical amplifier (~ 1 - 10 W) to extend transmission range and the receiver needs to have a very sensitive (quantum-limited) front-end to extract maximum communication performance.

This paper will review the status of these critical technologies and system architecture development. We will also point out where future technology development and new system architectures can provide large pay-offs in cost and performance. Finally we will give some notions of space network architectures and potential cost/performance that can lead to, in the future, more extensive macro-economic modeling of such systems as viable commercial ventures.

On the benefits of configurability in WDM Networks

Eytan Modiano
Massachusetts Institute of Technology
LIDS, 33-219
Cambridge, MA 02139
Modiano@mit.edu

Aradhana Narula-Tam
Massachusetts Institute of Technology
Lincoln Laboratory
Lexington, MA 02420
Arad@ll.mit.edu

Abstract

Configurable WDM components, such as frequency selective switches and tunable transceivers can be used to dynamically reconfigure the logical topology of a network in response to changes in traffic conditions. In this paper we study the potential benefit of such reconfiguration in both circuit and packet switched networks. In circuit switched networks, reconfiguration can be used to reduce call blocking probability, or equivalently, to increase network load for a given blocking probability. In packet switched networks, reconfiguration can be used to reduce network loads or delays. For both cases we present results on the potential benefits of reconfiguration in WDM ring networks.

I. Introduction

Wavelength Division Multiplexing (WDM) is emerging as a dominant technology for use in backbone networks. With WDM, the capacity of a fiber is significantly increased by allowing simultaneous transmission on multiple wavelengths (channels), each operating at the maximum electronic rate. Systems with between 40 and 80 wavelengths are presently being deployed for point-to-point transmission. With 10's of wavelengths per fiber and transmission rates of up to 10 Gbps per wavelength, capacities that approach a Tera-bit per second can be achieved. While such enormous capacity is very exciting, it also places a tremendous burden on the electronic switches and routers at each node that must somehow process all of this information. Fortunately, it is not necessary to electronically process all traffic entering and leaving each node. For example, much of the traffic passing through a node is neither sourced at that node nor destined to that node. To reduce the amount of traffic that must be electronically processed at intermediate nodes, future WDM systems will employ WDM Add/Drop multiplexers (WADMs), that allow each wavelength to either be dropped and electronically processed at the node or to optically bypass the node electronics.

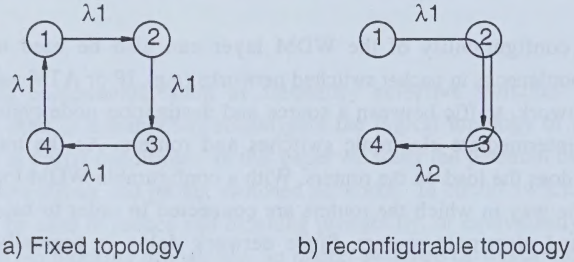
While much of today's circuit traffic is statically allocated, in the future there will be an increased demand for dynamically assigned (switched) high rate circuits. For high rate circuit switched traffic, where each circuit requires on the order of a full wavelength, configurable WADMs may be used to selectively add/drop only wavelengths carrying traffic that is either sourced or destined to a particular node. In this way, the configurable WDM optical layer is used to dynamically reconfigure the "logical" (i.e., electronic) topology of the network in response to new call arrivals. This can lead to a significant increase in the capacity that the network can support. With a fixed WDM layer, calls may be blocked due to a lack of electronic resources at intermediate nodes between the source and the destination. However, a configurable optical layer can be used to dynamically bypass the electronic layer on heavily loaded intermediate nodes and hence increase the traffic load that can be supported in the network. The potential gains in circuit capacity achievable with a configurable WDM physical layer are discussed in section II of this paper.

The configurability of the WDM layer can also be used to reduce electronic processing bottlenecks in packet switched networks (e.g., IP or ATM networks). In a packet switched network, traffic between a source and destination node typically goes through a number of intermediate electronic switches and routers. As the traffic in the network changes, so does the load on the routers. With a configurable WDM topology, it is possible to change the way in which the routers are connected in order to balance the load on the lightpaths and electronic routers. Since network delay is generally dominated by the maximally loaded link/router, reducing the load on this link can significantly improve network delay characteristics. This reconfiguration of the logical topology can be executed in response to changes in traffic conditions. In section III we discuss load balancing algorithms and the potential reduction in network load.

II. Circuit switched networks

In WDM networks, the physical topology consists of passive or configurable optical nodes interconnected by fiber links. The logical or virtual topology, seen by the electronic layer, consists of lightpath connections between nodes established by tuning the transmitter of one node and the receiver of another node to the same wavelength. In this way, WDM networks provide a way to interconnect electronic switches with high bandwidth bit pipes without dedicating a fiber pair between each pair of switches. Furthermore, the configurable nature of WDM also allows these high bandwidth pipes to be dynamically reconfigured in response to changes in traffic conditions.

Consider for example a network where each node is equipped with a single port (transmitter/receiver) and the fiber supports two wavelengths, $\lambda 1$ and $\lambda 2$. In a system with a fixed logical topology, the transmitters and receivers are fixed tuned to their respective wavelengths at system inception. With configurable WADMs and tunable transmitters and receivers, the logical topology can be reconfigured to accommodate changing patterns in externally offered traffic. In Figure 1a, we show a four node example where the fixed topology is a clockwise unidirectional ring. If a full-wavelength call is in progress from node 1 to node 3, and a call request arrives between nodes 2 and 4, then that request must be blocked. In a reconfigurable system, both calls can be supported as shown in figure 1b. Reconfiguring the logical topology allows traffic to optically bypass intermediate node electronics thereby minimizing the number of electronic ports necessary. In the example above, by reconfiguring the logical topology, the traffic from node 1 to node 3 optically bypasses node 2, thereby allowing the port at node 2 to carry traffic from node 2 to node 4. Similarly, the traffic from node 2 to node 4 optically bypasses node 3 whose port is busy with the connection from node 1 to node 3.



a) Fixed topology b) reconfigurable topology

Figure 1. Using WDM to reconfigure the electronic topology.

In a circuit switched network, reconfiguration increases the traffic load that the network can support for a given blocking probability. The benefits of reconfiguration can be quantified by comparing the blocking probabilities of fixed topology and configurable topology networks. Many researchers have studied blocking probabilities for WDM networks with and without wavelength changers [2-4]. This prior work, however, assumed that wavelengths were the most precious network resource. Therefore, in analyzing blocking probability previous researchers assumed that a call between a source and destination would be accepted if a wavelength was available between the source and destination, and blocked otherwise. This ignored the possibility of calls being blocked due to the lack of electronic resources. However, when considering a multi-hop circuit switched network, calls can be blocked even when wavelengths are available. A call may be blocked when ports on the source or destination nodes are occupied or when an intermediate node has no ports available, as demonstrated in the example. By reconfiguring the logical topology such that traffic optically bypasses intermediate nodes whenever possible, the number of ports required to sustain each call is minimized. In order to analyze the blocking probability in a system where both wavelengths and electronic ports are limited, a model for blocking probability that takes into account both restrictions is needed.

In [5] an approximate model for blocking probability in such a network was developed. The model developed in [5] assumes that calls arrive according to a Poisson process, utilize a full wavelength, and have exponential holding times. The model also assumes that, once placed, calls cannot be rearranged or rerouted. Prohibiting rerouting of existing calls eliminates the possibility of calls being adversely affected by reconfiguration. This model was used in figures 2 and 3 to compare the performance of fixed topology and configurable topology ring networks with 10 and 100 nodes, respectively. Figures 2 and 3 show the reconfiguration capacity gain, defined as the ratio of the traffic load that can be supported by a reconfigurable ring network to that of a fixed topology bi-directional ring network, at a blocking probability of 0.01. These graphs illustrate that the capacity gain due to reconfiguration is most significant when the ratio of wavelengths to ports per node (W/P) is large and the number of ports per node is small.

The results from [5] indicate that when the number of wavelengths is much larger than the number of available ports per node, a capacity gain on the order of N/2 can be obtained in a unidirectional ring, where N is the number of nodes in the network. In a bi-directional ring a factor of N/4 increase in capacity can be obtained. Intuitively, this can be explained by observing that when the number of wavelengths is much greater than the number of ports per node, the system is port limited. That is, calls are blocked only when there are no ports available to complete the call. With a fixed topology system configured as a unidirectional ring, each call takes an average of N/2 hops and hence N/2 ports (transmitters and receivers). With a configurable topology, each call only uses only one transmitter and one receiver. Therefore, a configurable system requires N/2 fewer transceivers per call, and can therefore support N/2 times the traffic of a fixed topology system. Similarly, in a bi-directional ring, each call takes an average of N/4 hops leading to an N/4 capacity gain when the topology is configurable. When the number of wavelengths approaches the number of ports per node, the benefits of reconfiguration are significantly diminished since both the configurable and fixed topology systems are wavelength limited, and many calls are blocked due to insufficient bandwidth availability. However, even in this wavelength limited case, a factor of two gain in capacity is achieved by the configurable system.

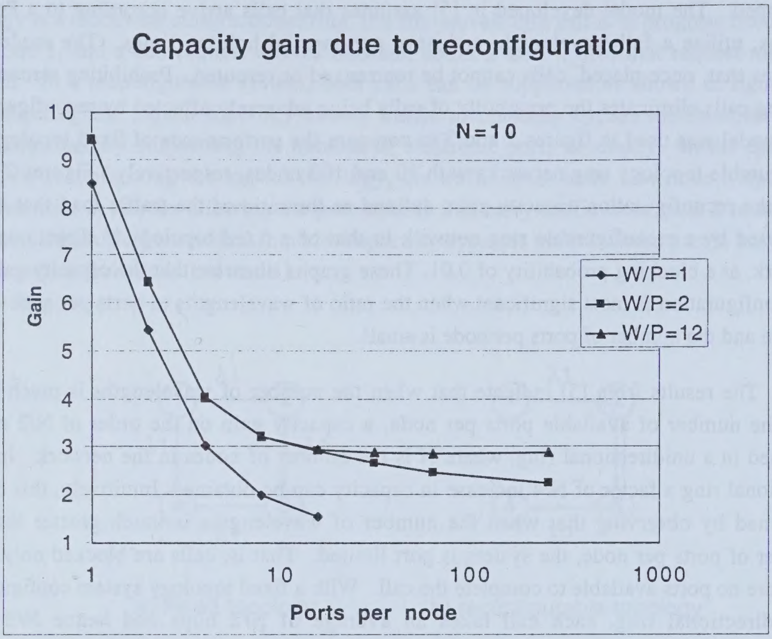


Figure 2: Capacity gain due to reconfiguration in a 10 node bi-directional ring.

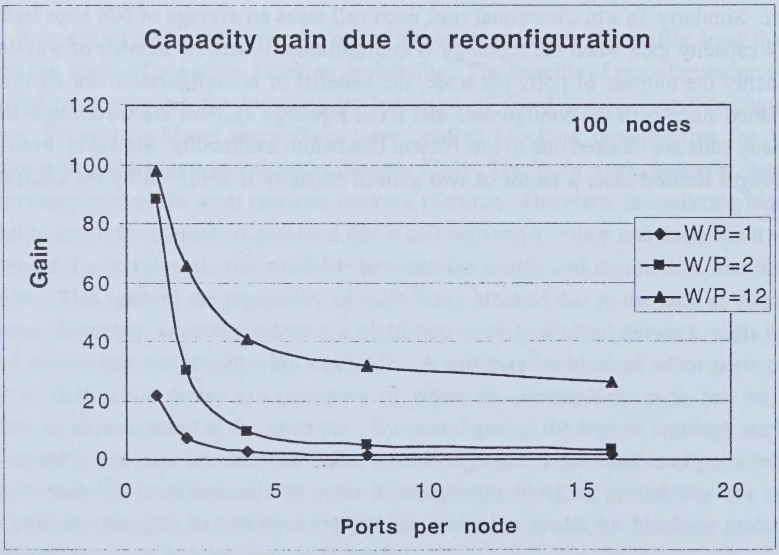


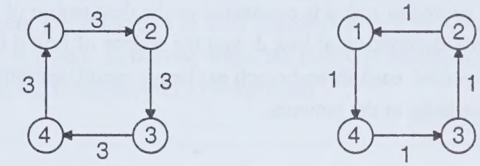
Figure 3: Capacity gain due to reconfiguration in a 100 node bi-directional ring.

III. Packet switched networks

The concept of reconfiguring the virtual topology of the network in response to changes in traffic conditions can also be applied to packet switched networks for the purpose of balancing the load on the electronic routers in the network. Consider, for example, the network shown in figure 4. Shown on the left (a), is the traffic matrix denoting the traffic requirement between the various nodes (i.e., one unit of traffic between node 1 and 4, etc.). When this matrix is routed on the clockwise ring shown in figure 8b, the resulting load on each link of the ring is 3 units of traffic. Using configurable WADMs and tunable transmitters and receivers, the logical topology of the ring can be rearranged, as shown on the right, so that a counter-clockwise ring results and the load on each link is only one unit of traffic. This reduces the load on each of the electronic routers by a factor of 3.

$$T = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

a) Traffic matrix



b) Fixed topology

c) Reconfigured topology

Figure 4: Reconfiguring the logical topology of the network to reduce the traffic load.

Since network delays are generally limited by the link/router carrying the maximum load, it is beneficial to reconfigure the logical topology to minimize the maximally loaded link. The problem of determining the optimal logical topology, however, is NP-complete [8]. Therefore, most prior work on topology design for WDM-based packet networks proposes heuristics for solving the optimal virtual topology design problem [6,7,10,11].

Once the new configuration is determined, the lightpaths from the old topology may be replaced by lightpaths for the new configuration. However, this process of reconfiguring to the new network topology can be disruptive to existing traffic. Traffic must be re-routed or buffered at each node while the transmitters and receivers are re-tuned. To minimize network disruption, methods of migrating to the optimal topology using a sequence of branch-exchanges (i.e., exchanging the source and destinations of two links in the network) have been developed [12]. Furthermore, it has been suggested that reconfiguration occur very infrequently, for example once or twice a day. Policies for reconfiguring when the benefits of reconfiguration outweigh the costs were examined in [9].

Reconfiguring the logical topology infrequently, however, may lead to logical topologies that are obsolete for much of the time they are in use. If traffic changes rapidly, it may not be possible, or desirable, to completely reconfigure the logical topology of the network with every perceived traffic change. Instead, a more gradual change in the topology may be desired. Migrating to the optimal topology using branch-exchange sequences as described in [10] minimizes the network disruption at each step. However, if traffic is rapidly changing, by the time that the new topology is implemented, the traffic may have changed. Furthermore, intermediate steps may leave the network temporarily disconnected.

In [13] a reconfiguration strategy is developed that updates the logical topology via small changes at regular time intervals. The small changes in the topology limit the disruption to the network, allowing reconfiguration to be employed more regularly and consequently tracking the changes in traffic patterns. For a network in which each node is equipped with a single transmitter and receiver, the logical topology is iteratively improved using a three-branch exchange mechanism. A three-branch exchange selects three links in the ring, for example links a, b, and d, shown in figure 5, and reorders the connections so that the source of link a is connected to the destination of link b, the source of link b is connected to the destination of link d, and the source of link d is connected to the destination of link a. Therefore, each three-branch exchange maintains network connectivity while disrupting only three links in the network.

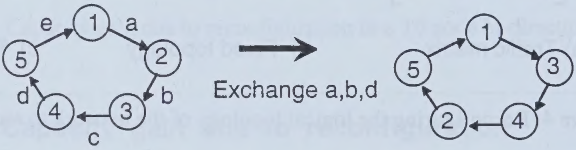


Figure 5: Three branch exchange sequence.

The algorithm in [13] performs a single three-branch exchange at regular time intervals (e.g., one per second). At each step, the algorithm searches for the three-branch exchange that maximally reduces the maximum link load. In order to evaluate the performance of the algorithm, two time-varying traffic models were utilized. In the first traffic model, the traffic between each node pair is independent and identically distributed (i.i.d) according to a uniform distribution. In the second traffic model, a number of clusters are generated (at random locations) where the traffic amongst nodes within a cluster is significantly greater than the traffic amongst non-cluster nodes. This clustered or hot spot traffic model represents the situation where a large amount of traffic may be flowing from a source node to several destination nodes (file server model) and also the scenario in which several source nodes are sending large amounts of traffic to a single destination. Furthermore, when the network nodes are aggregation points, clustered traffic is quite natural. Under clustered traffic, reconfiguration is expected to produce significant performance gains because the logical topology of the network can be tailored to the structured traffic pattern. This is in contrast to the i.i.d traffic model in which traffic between each pair of nodes is uncorrelated. To model the time-varying nature of traffic, the traffic evolves to a new random traffic matrix every D seconds.

Since the goal of reconfiguration is to reduce the maximum link load on the network, the performance of the algorithm can be measured in terms of the percent reduction in maximum load that it achieves over a fixed topology network. Figures 6 and 7 plot this reduction for i.i.d and clustered traffic respectively, assuming that the traffic evolves to a new random traffic matrix every ten seconds and that a single three-branch exchange is implemented every second. This corresponds to ten reconfiguration steps between two independent random traffic realizations. The figures show the percent reduction in maximum link load as a function of time. Also, shown in the figures is the reduction in maximum load that can be achieved if the optimal topology is utilized at all times. As can be seen from the figures, the algorithm tracks the optimal topology very closely. However, it is also clear that the gain due to reconfiguration is rather modest when the traffic is i.i.d. In fact, for i.i.d traffic the average reduction in load using the algorithm is 10% (compared to a maximum gain of 11% when the optimal topology is used). However, when the traffic is clustered, an average reduction in load of 30% is achieved (compared to 31% optimal).

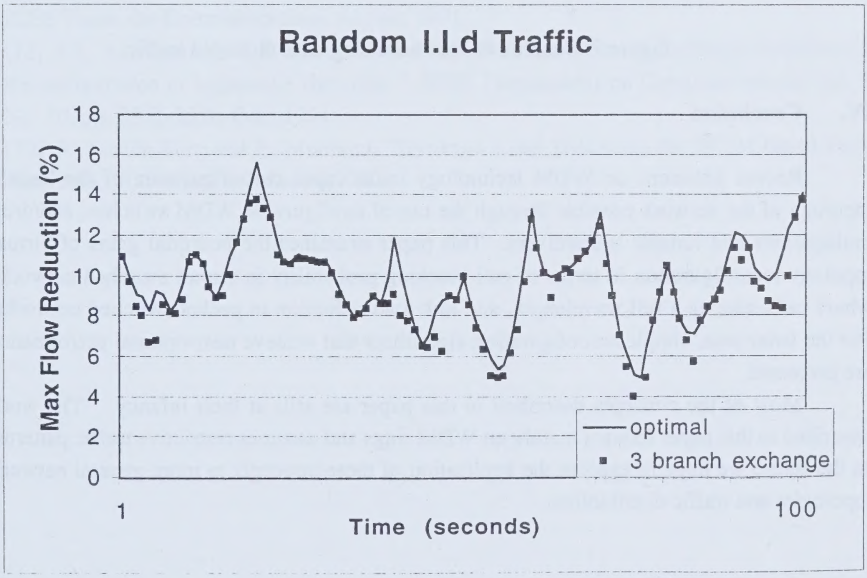


Figure 6: Benefits of load balancing with random i.i.d traffic.

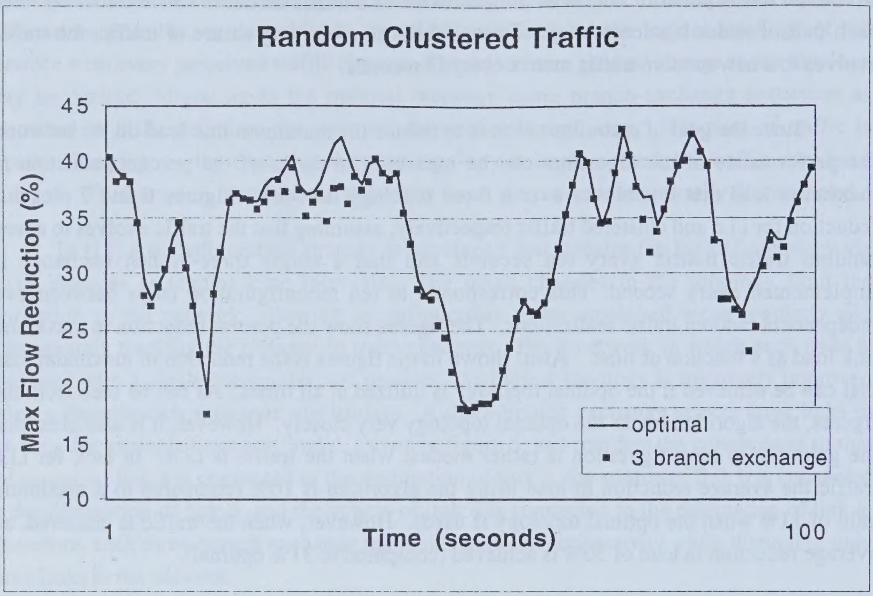


Figure 7: Benefits of load balancing with clustered traffic.

IV. Conclusion

Recent advances on WDM technology make rapid reconfiguration of the logical topology of the network possible through the use of configurable WDM switches, add/drop multiplexers and tunable transceivers. This paper examines the potential gains of virtual topology reconfiguration in terms of call blocking probability in circuit switched networks where calls take up a full wavelength, and link load reduction in packet switched networks. For the latter case, simple reconfiguration algorithms that achieve near-optimal performance are presented.

Most of the concepts described in this paper are still at their infancy. The work described in this paper focuses mainly on WDM rings and assumes restrictive traffic patterns. In the future we hope to explore the application of these concepts to more general network topologies and traffic distributions.

References

[1] S. Subramanian and R. Barry, "Wavelength Assignment in Fixed Routing WDM Networks,", Proc. of ICC '97, Montreal, CA, June, 1997.

[2] R. Barry and P. humblet, "Models of Blocking Probability in All-Optical Networks with and without Wavelength Changers," IEEE JSAC, Vol. 14, pp.858-867, 1996.

[3] Y. Zhu, G. N. Rouskas and H. G. Perros, "Blocking in Wavelength Routing Networks, Part I: The Single Path Case," Infocom '99, NY, NY, March, 1999.

[4] A. Birman, "Computing Approximate Blocking Probabilities in a Class of All-Optical Networks," IEEE JSAC, June, 1996.

[5] B. Schein and E. Modiano, "Increasing Traffic Capacity in WDM Ring Networks Via Topology Reconfiguration," CISS '99, March, 1999.

[6] J. Bannister, L. Fratta, and M. Gerla, "Topological Design of WDM Networks," Proc. Infocom '90, Los Alamos, CA, 1990.

[7] R. Ramaswami and K. Sivarajan, "Design of Logical Topologies for Wavelength Routed Optical Networks," IEEE JSAC, Vol. 14, No. 5, pp. 840-51, June, 1996.

[8] D. Bienstock and O. Gunluk, " Computational Experience with a Difficult Mixed-Integer Multicommodity Flow Problem," Mathematical Programming, Vol. 69, pp. 213-237, 1995.

[9] G. N. Rouskas and M. H. Ammar, "Dynamic Reconfiguration in Multi-hop Lightwave Networks," J. High Speed Networks, pp. 221-238, 1995.

[10] S. Banerjee and B. Mukherjee, "The Photonic Ring: Algorithms for Optimized Node Arrangements," Fiber and Integrated Optics, Vol. 12, No. 2, pp. 133-17, 1993.

[11] J.-F. P. Labourdette and A.S. Acampora, "Logically Rearangable Lightwave Networks," IEEE Trans. On Communications, August, 1991.

[12] A.S. Acampora, J.-F. P. Labourdette and G. W. Hart, "Branch-Exchange Sequences for Reconfiguration of Lightwave Networks," IEEE Transactions on Communications, Vol. 39, No. 10, pp. 2822-2832, Oct., 1994.

[13] A. Narula-Tam and E. Modiano, "Dynamic Load Balancing for WDM-based Packet Networks," Submitted to Infocom 2000.

Nonblocking WDM Networks with Fixed-Tuned Transmitters and Tunable Receivers[1]

Tao Lin and Galen Sasaki
University of Hawaii
2540 Dole St, Honolulu, HI 96822
{sasaki, tlin}@www-ee.eng.hawaii.edu

We consider *wavelength routed networks* which are *wavelength division multiplexed* (WDM) optical networks. These are *circuit-switched* networks, where a *circuit* is an end-to-end optical connection referred to as a *lightpath*. The links are assumed to be *bidirectional*, i.e., carry optical signals in both directions. Such links are realized by two fiber links, each carrying signals in opposite directions. The nodes of the network are composed of WDM multiplexers and demultiplexers, optical cross connects, and optical transmitters and receivers. The multiplexers and demultiplexers are connected to the outgoing and incoming fibers, respectively. The multiplexers and demultiplexers are also connected to the optical cross connect, that switches the individual optical signals between them and ultimately between the fibers. In this way, the node serves as a *wavelength selective switch*, i.e., it can independently switch signals at different wavelengths. The transmitters and receivers are also connected to the optical cross connect so that they may source and sink lightpaths. (Notice that our lightpaths are directed.) In addition, we allow *wavelength conversion* at intermediate nodes, where an incoming optical signal may be converted to an electronic one via a receiver and then converted to an optical signal again via any one of the transmitters. Here, we assume that the nodes have electronic cross connects to accomplish the electronic switching.

In [1], wavelength routed networks with ring, two-dimensional square grid, and hypercube topologies were investigated. In particular, the wavelength requirements were determined to insure no blocking of lightpaths given the traffic was *permutation traffic*, i.e., each node can source and sing at most one lightpath. In addition, it was assumed that transmitters and receivers were *tunable*, i.e., could change wavelength.

We consider the case where transmitters are *fixed-tuned*, i.e., the transmitters are given wavelengths at some initial time (perhaps at the time of fabrication) but do not change wavelengths thereafter. Presumably fixed-tuned transmitters will be cheaper and simpler to construct than tunable ones. Our aim is to determine the minimum number of transmitters and receivers at each node to get a reasonably good wavelength efficiency. We say that a nonblocking ring (resp., square grid) network with N nodes has *good wavelength efficiency* if it has $O(N)$ (resp., $O(\sqrt{N})$) wavelengths. This definition is motivated by the fact that for the ring (resp., grid) topology, there is a traffic pattern where there is a link with traffic load approximately $\frac{N}{4}$ (resp., $\frac{\sqrt{N}}{2}$) or more. This can be shown by having the traffic cross a topology (graph) bisection.

To be nonblocking, a network must have at least one transmitter-receiver pair per node. With just one transmitter-receiver pair per node, we have a lower bound on the number of wavelengths required for a nonblocking network stated in the next theorem. We will use the following definition. For a network topology $G = (V, E)$, let $f(G) = max_{S\subset V:|V|>c(S)} \frac{|V|-|S|}{c(S)}$, where $c(S)$ is the number of links between the subset of nodes S and $V\backslash S$. (In other words, $c(S)$ is the *capacity* of *cut* $(S, V\backslash S)$.) If there is no subset S satisfying $|S| > c(S)$ then we define $f(G) = 1$.

---

[1] This work was supported in part by the National Science Foundation under grant NCR-9612846

Theorem 1 *Consider a nonblocking network with topology $G = (V, E)$ with a transmitter-receiver pair at each node. In addition, the transmitters are fixed-tuned, and the receivers are tunable. Then the number of wavelengths W is at least $f(G)$.*

Proof. First, note that the network will not allow wavelength conversion because this will expend a transmitter-receiver pair at an intermediate node which in turn will lead to blocking at the node. Now suppose there are no subsets S such that $|S| > c(S)$. Then the lemma is trivially true because $f(G) = 1$.

Next, suppose there is some set S such that $|S| > c(S)$. Let S be such a set. Let $V' = V \backslash S$, and for each wavelength w, let V'_w be the nodes of V' that have transmitters fixed-tuned to w. Now $|V'_w| \leq c(S)$. Otherwise, we get blocking when $c(S) + 1$ nodes in V'_w need to be connected to $c(S) + 1$ nodes in S. There is blocking because there must be $c(S) + 1$ link disjoint lightpaths at wavelength w, but there are only $c(S)$ links connecting S to the rest of G. Now we have the following: $|V| - |S| \leq |V'| \leq \sum_w |V'_w| \leq \sum_w c(S)$ $\leq W \cdot c(S)$. Thus, $W \geq \frac{|V|-|S|}{c(S)}$. Since S was arbitrary, the lemma is true. □

The next corollary follows from the theorem for the square grid and ring. For the ring, S is three consecutive nodes. Then $|S| = 3$ and $c(S) = 2$. For the grid, S is the nodes in a 3×2 subgrid in the corner of the grid. Then $|S| = 6$ and $c(S) = 5$.

Corollary 1 *Consider a nonblocking network with N nodes and each node has a fixed-tuned transmitter and tunable receiver. If the topology is a ring then, for sufficiently large N, the number of wavelengths is at least $\frac{N-3}{2}$. If the topology is a square grid then, for sufficiently large N, the number of wavelengths is at least $\frac{N-6}{5}$.*

In [1], a nonblocking ring network was given that has one fixed-tuned transmitter and one tunable receiver per node, and $\lceil N/2 \rceil$ wavelengths. Thus, for ring networks, a single transmitter-receiver pair per node leads to good wavelength efficiency. For grids, the corollary implies that we need more than a single transmitter-receiver pair. The next theorem shows that two pairs per node are sufficent for good wavelength efficiency.

Theorem 2 *For every N that is the square of an integer, there is a rearrangeable nonblocking network that has a square grid topology, two transmitter-receiver pairs per node, fixed-tuned transmitters, tunable receivers, and $2\sqrt{N}$ wavelengths. (Recall, rearrangeable nonblocking means that existing lightpaths can be rearranged to accomodate new ones.)*

The following describes the grid network of the theorem. The wavelengths are divided exactly in half into two types, referred to as *access* and *primary* wavelengths. Each node has an access and primary wavelength that are distinct from the wavelengths of other nodes in the same row. Each lightpath is identified by the row of its source node and column of its destination node. To set up lighpaths in the network, each lightpath is given a primary wavelength that is distinct from the wavelengths of other lightpaths having the same row or column (this is possible as shown in [2]). Then a lightpath uses its source node's access wavelength to get to the node on its row with its primary wavelength. Then it uses the primary wavelength, to follow its row then column to its destination.

References

[1] C. Qiao and Y. Mei, "Off-line permutation embedding and scheduling in multiplexed optical networks with regular topologies," *IEEE/ACM Trans. Networking*, vol. 7, no. 2, pp. 241-250, April 1999.

[2] A. Youssef, "Off-line permutation scheduling in circuit-switched fixed routing networks," *Proc. SYmp. Frontiers of Massively Parallel Computation,* 1992, pp. 389-396.

On New Architectures for WDM Networks

Arunabha Sen * and Tarak Shah
Department of Computer Science and Engineering
Arizona State University
Tempe, AZ 85287, USA
Subir Bandyopadhyay
School of Computer Science
University of Windsor
Windsor, Ontario N9B 3P4, CANADA
Bhabani P. Sinha
Department of Advanced Computing and Microelectronics Unit
Indian Statistical Institute
Calcutta 700035, INDIA

Abstract

In this paper we propose two different architectures, the *Multimesh* and the *Incomplete Multimesh*, for use in wavelength division multiplexed optical networks. These architectures can be deployed as a physical topology for single hop or a logical topology for multihop networks. In terms of simplicity of interconnection and routing, these architectures are comparable to the regular mesh and the torus. However, the new architectures exhibit significantly superior topological properties than the mesh and the torus. For example, whereas a 2-dimensional $N \times N$ torus with $2N^2$ links has a diameter (distance between two nodes of the network that are farthest from each other) of N, a multimesh network with the same number of nodes and links has a diameter of $\sqrt{N}$. The Incomplete Multimesh can be constructed for any number of nodes and is incrementaly expandable. We also introduce a new metric, the *flow number*, to evaluate different topologies for lightwave application. A topology with a lower flow number is considered to be more attractive for lightwave application then the one with a higher flow number. We show that the flow number of a 2-dimensional $N \times N$ torus is $N^3/8$ whereas the flow number of a multimesh network with the same number of nodes is $N^{2.5}/2$.

Keywords: Optical networks, mesh, torus, flow number, multimesh, multihop networks.

---

*Corresponding author: Arunabha Sen, Telephone: 602-965-6153, Fax: 602-965-2751, e-mail: arunabha.sen@asu.edu

1 Introduction

Optical networks use interconnection of high-speed broadband fibers to transmit information. The stations, (also referred to as the end-nodes), in such a network are interconnected by optical fibers and routers. Wavelength Division Multiplexing (WDM) enables multiple communications at different wavelengths over a single fiber. Each wavelength on a given fiber is called a channel. These networks support *lightpaths* which are end-to-end communications connections passing through one or more fibers, using one WDM channel per fiber. Optical networks can be divided into two classes - *single hop* and *multihop* networks. An important objective in single hop network design is to minimize the total number of carrier wavelengths (channels) needed for communication between various end stations. An important requirement for high throughput in a multihop optical network is to ensure that the average delay due to buffering at the intermediate nodes is small. Simple interconnection topologies, such as the *ring* and the *mesh*, and more complex ones, such as the *shufflenet* and the *de Bruijn graph* have been proposed for single and multihop networks [7, 9, 12]. Ideally, a topology should be incrementally scalable, should have a simple routing strategy and should be able to communicate even in the presence of node (router) and link failure. In addition, the requirement for carrier wavelengths (channels) should be low in a single hop network and the average delay should be low in a multihop network. None of the proposed topologies satisfy all these orthogonal, if not conflicting, requirements [8, 9].

In a single hop network, a *lightpath* is a path from a source S to a destination D that uses the same wavelength for communication over the entire path. It may be noted that this path may pass through several routers on its journey from the source to the destination. If a lightpath from S to D passes through k intermediate routers on its journey from the source to the destination, then it has to reserve $k+1$ channels, one for each segment (fiber) in the path from S to D. Since two different lightpaths cannot be assigned the same WDM channel if they share the same fiber [8], the demand for the number of channels per fiber will be higher if many lightpaths have to share the same fiber. As the length of the source-destination path is bounded by the diameter, one might think that the demand for the number of channels per fiber will be low, if the network topology has a small diameter. However, this is not necessarily true. The reason for this is that even though the topology may have a small diameter, the interconnection pattern may force a large number of lightpaths to travel through a small number of fibers, thereby increasing the demand for the number of channels in those fibers.

In a multihop network, the communication delay may be proportional to the number of intermediate routers the packets have to pass through on their journey from the source to the destination. As a consequence, a topolgy with a small diameter appears to be quite attractive in such an environment.

In this paper we have proposed the use of the *multimesh* network for lightwave application. The multimesh architecture was originally proposed for parallel processing systems [1, 2, 3]. We have generalized this architecture to a new one called the *incomplete multimesh* that can be constructed for any number of nodes and is incrementally expandable. These architectures can be used both as a physical topology as well as a logical topology for optical networks. In terms of simplicity of interconnection and routing, these architectures are comparable to the regular mesh and the torus. However, they exhibit significantly superior topological properties than the mesh and the torus. For example, whereas a 2-dimensional $N \times N$ torus with $2N^2$ links has a diameter of N, a multimesh network with the same number of nodes and links has a diameter of $\sqrt{N}$. We have also introduced the notion of *flow number*, a new graph based metric to be used for network topology evaluation. This metric supplements the currently used metrics such as the diameter and the connectivity for network evaluation. The flow number is useful in determining the suitability of any topology for optical networks. A topology with a lower flow number is considered to be more attractive for lightwave application than the one with a higher flow number. We show that the flow number of a 2-dimensional $N \times N$ torus is $N^3/8$ whereas the flow number of a multimesh network with the same number of nodes is $N^{2.5}/2$.

2 The Multi-Mesh and Incomplete Multi-Mesh Architectures

The Multi-Mesh (MM) interconnection network was originally proposed for parallel processing in [1]. Since then researchers have developed techniques for efficiently carrying out several different computational tasks on this network [2, 3]. The interconnection pattern in this topology is an extension of the simple mesh connection. In an $n \times n$ mesh, the processors are arranged in n rows and n columns. The MM uses this as a building block for the construction of the network. The idea is to use n^2 such blocks arranged again in n rows and n columns. Thus a MM network has exactly n^4 nodes. Each of these n^4 nodes are identified with a four tuple label (α, β, x, y). The first two, (α, β), identifies the block and the last two, (x, y), identifies the node within a block. Each of these co-ordinates can take a value between 1 and n (both inclusive).

The $n \times n$ nodes within each block are connected as a regular two dimensional mesh. The interblock connections are made using the following rules:

1.) $\forall \beta, 1 \leq \beta \leq n$, the node $(\alpha, \beta, 1, y)$ is connected to the node (y, β, n, α) where $1 \leq y, \alpha \leq n$,

2.) $\forall \alpha, 1 \leq \alpha \leq n$, the node $(\alpha, \beta, x, 1)$ is connected to the node (α, x, β, n) where $1 \leq x, \beta \leq n$,

It may be observed that the interconnection rules given above generates a regular topology where each node is connected to exactly four edges. A partially completed MM network with eightyone nodes is shown in figure 1.

The number of nodes in a $N \times N$ MM network is always n^4. This restriction may be acceptable in a parallel processing environment where the number of nodes do not change very often, but may not be acceptable in its use in a local, metropolitan or wide area network environment where the number of nodes change quite frequently. We propose a new variation of the MM interconnection topology where we relax the MM topology requirement that the number of nodes always be n^4. This variation of MM, called *Incomplete Multi-Mesh* (IMM) can be constructed for any number of nodes. The construction mechanism is simple. One needs to specify the total number of nodes P and the block size n. Then the IMM will have $m = \lceil P/n^2 \rceil$ blocks. The blocks will be arranged in rows and columns as in MM. There will be $\lceil m/n \rceil$ rows, with each row having at most n blocks. The last row of blocks may be paritially complete with $mod(m, n)$ blocks. A complete block will have $n \times n = n^2$ nodes. A partially complete block with

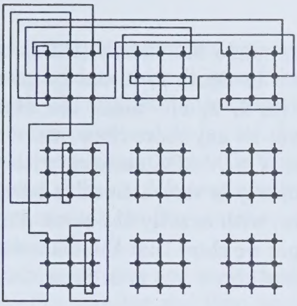


Figure 1: Multi-Mesh Network with 81 nodes (Not all interblock links are shown for clarity)

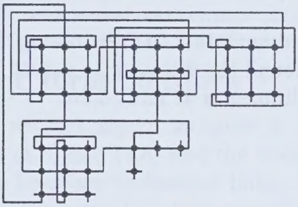


Figure 2: Incomplete Multi-Mesh Network with 40 nodes

q nodes will have $\lfloor q/n \rfloor$ complete rows with each row having exactly n nodes. The last row within a block may be paritially complete with $mod(q,n)$ nodes. An IMM with forty nodes is shown in figure 2. The main advantages of the IMM is that not only does it allow the construction of such a network for any number of nodes, it is also *incrementaly expandable* in the sense that an P node network can easily be augmented to an $P+1$ node network. This incremental expansion possibility of IMM makes it very attractive for use in a LAN/MAN/WAN environment where the number of nodes undergo frequent changes.

3 The Connectivity and Diameter of Multi-Mesh and Incomplete Multi-Mesh Architectures

A complete MM network with n^4 is a regular graph with node degree four. The *node connectivity* between a pair of nodes in a graph is the maximum number of *node disjoint* paths between these nodes. The *node connectivity* of a graph is the minimum of the node connectivity over all pairs of nodes. Since the degree of each node in a MM is four, it is clear that the upper bound of connectivity of MM is four. In this section, we show that in a MM network this upper bound is indeed realized, i.e., the connectivity of MM is four.

Theorem 1 *The connectivity of Multimesh network is four.*

Proof: The proof of the theorem is omitted due to page limitation. It can be found in [10].

The connectivity of an Incomplete Multimesh (IMM) can be as low as one. In the forty node IMM network shown in figure 2, it can be seen that the node (2, 2, 2, 1) is attached to only one node (2, 2, 1, 1). Since the degree of this node is one, the connectivity of this graph cannot be any more than one.

The diameter of a complete $N \times N$ MM network with $N = n^2$ nodes was shown to be $2n = 2N^{1/2}$ in [1]. This property is very attractive because, a comparable network - a two dimensional $N \times N$ torus, with exactly the same number of nodes and edges will have a diameter N. In this paper we show that the diameter of an IMM network with P nodes $(1 \leq P \leq n^4)$ and blocks of size $n \times n$ is at most $4n$.

Theorem 2 *The diameter of an Incomplete Multimesh network at most* $4n$.

Proof: The proof of the theorem is omitted due to page limitation. It can be found in [10].

Note: In our extensive experimentation to determining the diameter of an incomplete multimesh network, we never found the diameter to exceed $3.5n$. We *conjecture* that the diameter of an incomplete multimesh is at most $3.5n$.

4 Flow Number

The metrics like the *connectivity* and the *diameter* of a graph $G = (V, E)$ are used to evaluate the relative merits of different topologies. However, these metrics are unable to capture one important aspect of a network topology. Suppose $d_{i,j}$ is the demand for data stream from node i to node j, $(1 \leq i, j \leq n)$. For example, $d_{i,j}$ may be 1Mbps, implying that 1M bits has to flow from node i to node j per second. Suppose that a demand vector $D = [d_{1,1}, \ldots, d_{n,n}]$ is specified $(d_{i,i} = 0\ \forall i, 1 \leq i \leq n)$. The traffic flow from node i to node j may be routed in many different ways. However, no matter how it is routed, such a flow may be impossible if the capacity (bandwidth) associated with the links is below a certain threshold value. Assuming that the capacity of all the links are identical, we would like to find this minimum threshold capacity of the links such that the demand D can be met through some routing. We refer to this minimum threshold capacity on the links that is able to sustain a traffic flow $d_{i,j} = 1\ \forall i, j,\ i \neq j$, as the *flow number* of the graph $G = (V, E)$.

Formally, the flow number $fn(G)$ of a graph $G = (V, E), (|V| = n, |E| = m)$ may be defined as follows: Suppose that $R_1, \ldots, R_k$ is the set of all possible ways of routing the traffic from the set of sources to the set of destinations. The traffic flow is $d_{i,j} = 1\ \forall i, j,\ i \neq j$. We refer to the traffic from a source i to a destination j as a data stream or a *flow*. Consider a $m \times k$ traffic matix T whose (i, j)-th entry, $T[i, j]$, indicates the number of data streams (or flows) through the edge (link) e_i of the graph $G = (V, E)$ under the routing scheme R_j. Then the flow number of $G = (V, E)$ is

$$fn(G) = \min_{1 \leq j \leq k} [\max_{1 \leq i \leq m} T[i, j]]$$

It may be noted that the flow number of a graph can be computed using the techniques known for the *multicommodity flow* problem [5]. In a sense, it can be viewed as the dual of the multicommodity flow problem. In the multcommodity flow problem, the capacity of the links is known and the objective is to find the maximum flow that can be attained, subject to the link capacity constraints. In flow number computation, traffic demand for all source-destination pair is known and the objective is to find the minimum link capacity that will be able to satisfy all the demands. It is well known that the multicommodity flow problems are NP-complete [4]. Approximation algorithms for these problems with a guaranteed performance bound is presented in [6, 11]. In the following two subsections, we show that if the graph has a regular structure, the computation of the flow number of a graph is not necessarily all that difficult. We compute the flow number of a $N \times N$ torus and multimesh network and show that the flow number of the torus is $N^3/8$ whereas the flow number of the multimesh is $N^{2.5}/2$.

We compute the flow numbers of torus and multimesh in the following way: First we compute a lower bound on the flow number and then we compute an upper bound on the flow number and finally we show that these two are equal. Each undirected edge of both the torus and the multimesh is replaced by two oppositely directed edges. We compute the minimum capacity on these directed edges that will be able to satisfy the traffic demand of one unit for every source-destination pair ($d_{i,j} = 1 \ \forall i, j, \ i \neq j, d_{i,i} = 0 \ \forall i$).

4.1 Flow Number of Torus Network

Consider the $N \times N$ torus network shown in figure 3. It may be observed that only the links between the nodes (1,4) and (1,5) and the nodes (1,1) and (1,8) are shown as the directed links, all other links are undirected links. This is done only to keep the diagram uncluttered. For the purpose of analysis we will consider an undirected edge as comprising of two oppositely directed edges. For simplicity of analysis, we assume that $N = 4m$ for some positive integer m.

Lemma 3 *The lower bound on the flow number of a torus network is* $N^3/8$.

Proof: Every node in this $N \times N$ torus network has to transfer one unit of data to every other node. Consider the vertical line AB that divides the torus in two equal halves. Therefore, each of the $N^2/2$ source nodes on the left half has to transfer data to each of the $N^2/2$ destination nodes on the right half of the torus. The links *cut* by the line AB has to carry all these $N^2/2 * N^2/2 = N^4/4$ traffic from the left half of the torus to the right half. There are exactly $2N$ directed edges (links) connecting the nodes on the left half to the nodes on the right half. (Each of the N rows of the torus has two such links; in figure 3, on the first row these links are from the node (1,1) to (1,8) and the node (1,4) to (1,5)). Since $2N$ links has to carry $N^4/4$ flows, the maximum of the minimum flows on these $2N$ links must be $(N^4/4)/2N = N^3/8$. This is a *lower bound* on the flow number of a $N \times N$ torus.

Lemma 4 *The upper bound on the flow number of a torus network is* $N^3/8$.

Proof: The upper bound is established by first describing a routing algorithm. We show that if this algorithm is executed, no directed edge (link) will be required to carry more than a certain number of data streams or flows. This establishes an upper bound on the flow number.

As indicated earlier, we assume that $N = 4m$ for some positive integer m. Assume that the torus is divided into two halves by the line AB as shown in figure 3. There are $N^2/2 = 8m^2$ nodes on each of these two halves. The $8m^2$ nodes are distributed in $4m$ rows, with each row having $2m$ nodes. Consider another cut line CD shown in figure 3 that divides the left half into two parts. The $2m$ nodes on the first row in the left half is divided into two equal parts of m modes each by this line. These $2m$ nodes (the nodes $(1,1),\ldots,(1,4)$ in figure 3) have to transfer one data stream to the nodes on the right side (the nodes $(1,5),\ldots,(1,8)$ in figure 3). The links $(1,1) \rightarrow (1,8)$ and $(1,4) \rightarrow (1,5)$ can be used for this purpose. Therefore, the traffic on any one of these two links will be $m * 2m = 2m^2$. The routing algorithm will route the traffic in such a way that the traffic from the nodes $(i, m+1),\ldots,(i, 2m), 1 \leq i \leq 4m$ will use the link $(j, 2m) \rightarrow (j, 2m+1)$, to travel to the nodes $(j, 2m+1),\ldots,(j, 4m), 1 \leq j \leq 4m$. For example, the path taken by the traffic to reach the destination node $(j, 4m)$ from the source node $(i, m+1)$ will be as follows: $(i, m+1) \rightarrow (i+1, m+1) \rightarrow \cdots \rightarrow (j, m+1) \rightarrow (j, m+2) \rightarrow \cdots \rightarrow (j, 2m) \rightarrow (j, 2m+1) \rightarrow (j, 2m+2) \rightarrow \cdots \rightarrow (j, 4m)$. The traffic from the nodes $(i,1),\ldots,(i,m), 1 \leq i \leq 4m$ will use the link $(j,1) \rightarrow (j,4m), 1 \leq i,j \leq 4m$ to travel to the nodes $(j, 2m+1),\ldots(j, 4m)$. Therefore the total traffic on the links, $(j,1) \rightarrow (j,4m)$ and $(j,2m) \rightarrow (j,2m+1)$ for all j, $1 \leq j \leq 4m$ will be $2m^2 * 4m = 8m^3 = N^3/8$. Since, the torus has a completely symmetric structure, no directed edge will carry a traffic that is higher than this value. Thus $N^3/8$ is the upper bound on the flow number of the torus network.

Theorem 5 *The flow number of a* $N \times N$ *torus is* $N^3/8$.

Proof: Follows from lemmas 1 and 2.

4.2 Flow Number of Multimesh Network

Consider the $N \times N$ Multimesh network shown in figure 1. For the purpose of analysis we will consider an undirected edge as comprising of two oppositely directed edges.

Lemma 6 *The lower bound on the flow number of a Multimesh network is* $N^{2.5}/8$.

Proof: As shown in figure 4, the nodes in the $N \times N$ multimesh are arranged in blocks of smaller meshes of size $n \times n$, where $n = \sqrt{N}$. Thus, the total number of nodes is n^4 distributed in n^2 blocks with each block having n^2 nodes. Every node in this $N \times N$ multimesh network has to transfer one unit of data to every other node. Consider the vertical line AB that divides the multimesh in two equal halves. Therefore, each of the $N^2/2 = n^4/2$ source nodes on the left half has to transfer data to each of the $N^2/2 = n^4/2$ destination nodes on the right half of the multimesh. Since each block has exactly n^2 nodes, the number of blocks on both sides of the cut line AB is $n^2/2$. The number of rows of blocks on each sides is n. Therefore, the number of blocks in each row of blocks on each side of the line AB is $n/2$. Each block on the i-th row of blocks on the left half is connected to each block on the i-th row of blocks on the right half by two links, $(1 \leq i \leq n)$. Thus the total number of links that can carry traffic from the i-th row blocks on the left half to the i-th row blocks on the right half is $n/2 * n/2 * 2 = n^2/2$. The total number of rows of block is n. Thus the total number of links that can carry traffic from the left half of the network to the right half is $n * n^2/2 = n^3/2$.

These links *cut* by the line AB has to carry all these $N^2/2 * N^2/2 = N^4/4 = n^8/4$ traffic from the left half of the multimesh to the right half.

Since $n^3/2$ links has to carry $n^8/4$ flows, the maximum of the minimum flows on these $n^3/2$ links must be $(n^8/4)/(n^3/2) = n^5/2 = N^{2.5}/2$. This is a *lower bound* on the flow number of a $N \times N$ multimesh.

Lemma 7 *The upper bound on flow number of a Multimesh network is $N^{2.5}/8$.*

Proof: The upper bound is established by first describing a routing algorithm. We show that if this algorithm is executed, no directed edge (link) will be required to carry more than a certain number of data streams or flows. This establishes an upper bound on the flow number.

As indicated earlier, the nodes in the $N \times N$ multimesh are arranged in n^2 blocks of n^2 nodes each, where $n = \sqrt{N}$. The blocks are arranged in n rows and n columns. Each node in the multimesh interconnection is identified with a four tuple label (i, j, k, l). The first two (i, j) identifies the block and the last two (k, l) identifies the node within a block. Due to the construction rules of the multimesh, any two blocks that belong to the same row or same column form a *cycle*, in the sense that these two blocks are connected by two links. That is two blocks (i, j) and (i, k) have the following two links connecting them: $(i, j, k, 1)$ to (i, k, j, n) and (i, j, k, n) to $(i, k, j, 1)$.

Every node in this $N \times N$ multimesh network has to transfer one unit of data to every other node. Suppose the following routing technique is used: Consider a flow from the node $(i1, j1, k1, l1)$ to $(i2, j2, k2, l2)$. The flow first has to travel to the block identified by $(i2, j2)$ from the block $(i1, j1)$. The routing technique tries to match *first the row and then the column.* That is to travel from $(i1, j1)$ to $(i2, j2)$, first it travels to the intermediate block $(i2, j1)$ and then to the block $(i2, j2)$. Within a block also the routing algorithm follows the same *row first* policy. Suppose a flow has to travel from the node $(i1, j1, k1, l1)$ to $(i2, j2, k2, l2)$. According to the *row first* policy, the flow should travel from the $(i1, j1)$ block to the $(i2, j1)$ block first. The two nodes on the horizontal (vertical) boundary of a block responsible for communicating with other blocks in the network are referred to as the *communicating nodes.* Two communicating nodes on the horizontal boundary of the $(i1, j1)$ block is connected to two corresponding nodes on the horizontal boundary of the $(i2, j1)$ block. According to the routing technique considered here, the flow originating from the node $(k1, l1)$ within the $(i1, j1)$ block traverses to the *nearest* communicating node on the boundary of the $(i1, j1)$ block to leave that block and to enter the $(i2, j1)$ block.

The traffic that flows through the network can be divided into two classes: *inter-block traffic* and *intra-block traffic.* We first analyze the inter-block traffic and later we analyze the intra-block traffic.

Inter-block Traffic Analysis

As indicated earlier, the n^2 blocks of a multimesh network are arranged in n rows and n columns. Suppose the blocks are numbered from (1, 1) through (n, n). Consider the (i, j)-th block with n^2 nodes. The inter-block traffic that enters (or leaves) the (i, j)-th block can be divided into two classes: (A) The traffic that either originates or terminates at the (i, j)-th block, (B) The traffic that uses the (i, j)-th block as an intermediate node to get to the destination.

Class A: Originating or Terminating Traffic at the (i, j)-th block:

The (i, j)-th block of the multimesh network has n^2 nodes. The remaining $n^4 - n^2$ nodes of the network have data streams for each node of the (i, j)-th block. Similarly, each node of the (i, j)-th block have data streams for each of the remaining $n^4 - n^2$ nodes of the network. Thus, the total traffic originating from the other blocks and terminating in the (i,j)-th block is $(n^4 - n^2)n^2$. The same number of data streams originate at the (i,j)-th block and terminate at the other blocks. Like any other block of the network, the (i,j)-th block has $4(n-1)$ incoming (outgoing) links. Traffic from the blocks $(s,t), 1 \leq s,t \leq n, t \neq j$ will enter the block (i,j) through the nodes $(i,j,t,1)$ and the node (i,j,t,n). Thus the number of data streams on these two incoming links will be $n^2 * n^2 * n/2 = n^5/2 = N^{2.5}/2$. The outgoing data streams on these links will also be the same.

Traffic from the blocks $(s,j), 1 \leq s \leq n, s \neq i$ will enter the block (i,j) through the nodes $(i,j,1,s)$ and the node (i,j,n,s). Similarly, the traffic to the blocks $(s,j), 1 \leq s \leq n, s \neq i$ will leave the (i,j)-block through the nodes $(i,j,1,s)$ and the node (i,j,n,s). Thus the incoming (outgoing) traffic on these links will be $(n^2 * n^2 * (n-1))/2(n-1) = n^4/2$.

Class B: Transit Traffic at the (i,j)-th block

The traffic from the blocks $(s,j), 1 \leq s \leq n, s \neq i$ to the blocks $(i,t), 1 \leq t \leq n, t \neq j$ will enter the block (i,j) through the nodes $(i,j,1,s)$ and (i,j,n,s) and will leave the block (i,j) through the nodes $(i,j,t,1)$ and (i,j,t,n). Thus the total transit traffic on these incoming and outgoing links will be $(n^2 * n^2 * (n-1)^2)/2(n-1) = n^5/2 - n^4/2$.

Thus the total incoming (outgoing) traffic at the nodes $(i,j,1,s)$ and (i,j,n,s), $((i,j,t,1)$ and $(i,j,t,n))$ will be $n^4/2 + n^5/2 - n^4/2 = n^5/2 = N^{2.5}/2$.

Thus, it can be seen that the number of data streams on the inter-block links never exceeds the lower bound value of $N^{2.5}/2$.

Intra-block Traffic Analysis

In this part we consider the traffic carried by one directed edge (link) within a block. We consider the link $(i,j,k,l) \to (i,j,k,l+1)$, $1 \leq i,j,k,l+1 \leq n$ as shown in figure 5. The traffic carried by this link can be divided into four different categories:
(i) *Incoming Traffic:* Traffic originating in the block (s,t), $1 \leq s,t \leq n$, $s \neq i$, $t \neq j$ and terminating in the block (i,j).
(ii) *Outgoing Traffic:* Traffic originating in the block (i,j) and terminating in the block (s,t), $1 \leq s,t \leq n$, $s \neq i$, $t \neq j$.
(iii) *Transit Traffic:* Traffic neither originating nor terminating in the block (i,j) but traversing through the block (i,j) as an intermediate block.
(iv) *Internal Traffic:* Traffic originating and terminating in the block (i,j).

Case (i): Incoming Traffic on the link $(i,j,k,l) \to (i,j,k,l+1)$, $1 \leq i,j,k,l+1 \leq n$

The link $(i,j,k,l) \to (i,j,k,l+1)$ is shown in figure 5. The incoming traffic enters the block (i,j) through the communicating nodes on either the horizontal or the vertical boundary. The nodes $(i,j,p,1)$ and (i,j,p,n), $1 \leq p \leq n$, $p \neq j$ will have incoming traffic from the blocks (r,p), $1 \leq p,r \leq n$, $p \neq j$. Half the traffic from the block (r,p) will enter the block (i,j) through the node $(i,j,p,1)$ and the other half through the node (i,j,p,n). The total traffic (flow) from the block (r,p) to the block (i,j) is $n^2 * n^2 = n^4$. Half of this traffic, i.e., $n^4/2$ will enter the block (i,j) through the node $(i,j,p,1)$. This traffic will be divided into n equal parts to reach the destinations on n different rows of the (i,j)-th block. Thus the part of this traffic on the k-th row is $n^3/2$. Each node on this row consumes n^2 amount of the flow before it reaches the node (i, j, k, l). Therefore the part of this traffic on the link $(i, j, k, l) \rightarrow (i, j, k, l+1)$ is at most $n^3/2 - ln^2$. Thus, the total traffic from the block (r, p) on the link $(i, j, k, l) \rightarrow (i, j, k, l+1)$ is $(n^3/2 - ln^2) * (n^2 - n)$, for all possible values of r and p, $(1 \leq p, r \leq n,\ p \neq j)$.

The nodes $(i, j, 1, p)$ and (i, j, n, p), $1 \leq p \leq n,\ p \neq i$ will have incoming traffic from the blocks (r, j), $1 \leq r \leq n,\ r \neq i$. Half the traffic from the block (r, j) will enter the block (i, j) through the node $(i, j, 1, p)$ and the other half through the node (i, j, n, p). The total traffic (flow) from the block (r, j) to the block (i, j) is $n^2 * n^2 = n^4$. Half of this traffic, i.e., $n^4/2$ will enter the block (i, j) through the node $(i, j, 1, p)$. This traffic will be divided into $n/2$ equal parts to reach the destinations on n different rows of the (i, j)-th block. Thus the part of this traffic on the k-th row is n^3. Each node on this row consumes n^2 amount of the flow before it reaches the node (i, j, k, l). Therefore the part of this traffic on the link $(i, j, k, l) \rightarrow (i, j, k, l+1)$ is $n^3 - ln^2$. The link $(i, j, k, l) \rightarrow (i, j, k, l+1)$ will carry traffic from the block (r, j) as long as $1 \leq r \leq l$. Thus the total traffic from the block (r, j) on the link $(i, j, k, l) \rightarrow (i, j, k, l+1)$ is $l(n^3 - ln^2)$, for all possible values of r $(1 \leq r \leq l)$.

Thus the total incoming traffic on the link $(i, j, k, l) \rightarrow (i, j, k, l+1)$ is $(n^3/2 - ln^2) * (n^2 - n) + l(n^3 - ln^2) = n^5/2 - (l + 1/2)n^4 + 2ln^3 - l^2n^2$.

Case (ii): Outgoing Traffic on the link $(i, j, k, l) \rightarrow (i, j, k, l+1),\ 1 \leq i, j, k, l \leq n$

In the following, we present the analysis for the case $l \leq n/2$. The analysis is similar for the case when $l > n/2$ and is not presented here. The outgoing traffic on the link $(i, j, k, l) \rightarrow (i, j, k, l+1)$, $1 \leq i, j, k, l+1 \leq n$ when $l \leq n/2$ is zero unless $k = 1$ or $k = n$. We consider the case when $k = n$. In this case only the nodes (i, j, x, y), $n/2 < x \leq n$, $1 \leq y \leq l$ will be using the link $(i, j, k, l) \rightarrow (i, j, k, l+1)$ to transmit data to the $(l+1, z)$-th $(1 \leq z \leq n)$ block in the network. Thus the total traffic on the $(i, j, k, l) \rightarrow (i, j, k, l+1)$ will be at most $n/2 * l * n * n^2 = ln^4/2$.

Case (iii): Transit Traffic on the link $(i, j, k, l) \rightarrow (i, j, k, l+1),\ 1 \leq i, j, k, l+1 \leq n$

As in the previous case we consider the case $l \leq n/2$. When $l \leq n/2$, then the transit traffic through the link $(i, j, k, l) \rightarrow (i, j, k, l+1)$ is zero, beacuse the routing algorithm uses the *row first* policy. If $l > n/2$, then the traffic is non-zero and for the mathematical expression for the traffic the reader is referred to [10].

Case (iv): Internal Traffic on the link $(i, j, k, l) \rightarrow (i, j, k, l+1),\ 1 \leq i, j, k, l+1 \leq n$

The internal traffic on the link $(i, j, k, l) \rightarrow (i, j, k, l+1)$, $1 \leq i, j, k, l+1 \leq n$ is $nl(n - l)$.

Thus the total traffic on the link $(i, j, k, l) \rightarrow (i, j, k, l+1)$, $1 \leq i, j, k, l+1 \leq n$ is

$$n^5/2 - (l + 1/2)n^4 + 2ln^3 - l^2n^2 + ln^4/2 + 0 + ln^2 - l^2n$$

which after simplification is

$$n^5/2 - (l+1)n^4/2 + 2ln^3 - (l^2 - l)n - l^2n$$

The above expression is less than equal to $n^5/2 = N^{2.5}$ if $n \geq 4$. This establishes an upper bound on the intra-block traffic.

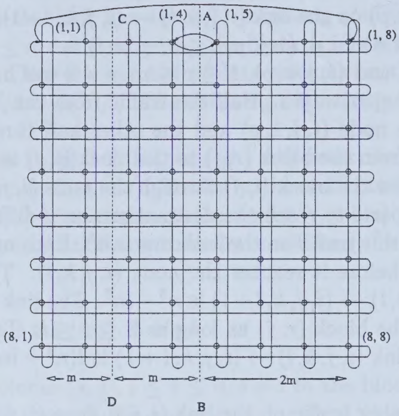


Figure 3: Torus Network

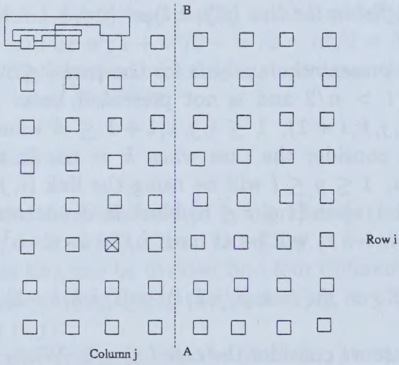


Figure 4: Blocks in a Multimesh Network

Theorem 8 *The flow number of a $N \times N$ multimesh is $N^{2.5}/2$.*

Proof: Follows from lemmas 3 and 4.

5 Conclusion

In this paper we have proposed two different architectures for wavelength division multiplexed optical networks. In addition, we have also introduced a new metric for the evaluation of interconnection networks for lightwave applications. Using previously known metrics, such as the connectivity and the diameter, as well as the newly introduced metric, the flow number, we have show n that the new architectures are more attractive than an architecture like the torus for application in the optical network domain.

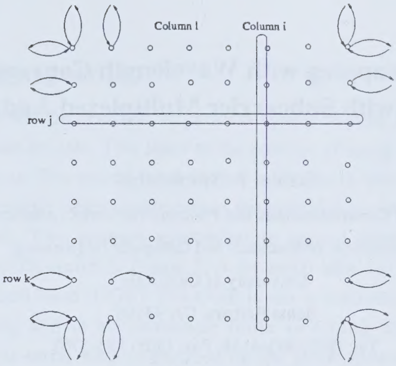


Figure 5: Nodes in one block of a Multimesh Network

References

[1] D. Das and B. P. Sinha, "Multi-Mesh - An efficient topology for parallel processing," *Proc. of 9th Int. Parallel Processing Symp.*, Santa Barbara, California, pp. 17-21, April 1995.

[2] M. De, B. Kundu and B. P. Sinha, "Wormhole routing for complete exchange in Multi-Mesh," *Proc. of 4th Int. Conf. on High Performance Computing*, Bangalore, India, December 1997.

[3] M. De, D. Das, M. Ghosh and B. P. Sinha, "An efficient sorting algorithm on the Multi-Mesh network," *IEEE Transactions on Computers*, vol. C 46, no. 10, pp. 1132-1137, 1997.

[4] M.R. Garey and D.S. Johnson, Computers and Intractability: A Guide to the Theory of NP-Completeness. W.H. Freeman Publishers, San Francisco, 1978.

[5] T. C. Hu, Integer Programming and Network Flows, Addison Wesley Publishing Company, 1970.

[6] P. Klein, S. Plotkin, C. Stein, E. Tardos, "Faster approximation algorithms for the unit capacity concurrent flow problem with applications to routing and finding sparse cuts", *SIAM Journal of Computing*, vol. 23, no. 3, pp. 466-487, 1994.

[7] M. A. Marsan, A. Bianco,E. Leonardi and F. Neri, "Topologies for Wavelength-Routing All-Optical Networks", *IEEE/ACM Transactions on Networking*, vol. 1, pp. 534-546, 1993.

[8] B. Mukherjee, Optical Communication Networks, McGraw-Hill, 1997.

[9] R. Ramaswami and K. Sivarajan, Optical Networks: A Practical Perspective, Morgan Kaufmann Publishers Inc, 1998.

[10] A. Sen, T. Shah, S. Bandyopadhyay and B.P. Sinha, "On new architectures for WDM networks", *Techinal Report, Department of Computer Science and Engineering, Arizona State University*, 1999.

[11] F. Shahrokhi and D.W. Matula, "The maximum concurrent flow problem", *Journal of the Association of Computing Machinery*, vol. 37, no. 2, pp. 318-334, 1990.

[12] K. Sivarajan and R. Ramaswami, "Lightwave networks based on de Bruijn graphs," *IEEE/ACM Transactions on Networking*, vol. 2, no. 1, pp. 70-79, 1994.

All-Optical Label Swapping with Wavelength Conversion for WDM-IP Networks with Subcarrier Multiplexed Addressing

Daniel J. Blumenthal
Optical Communications and Photonic Networks Laboratory
Department of Electrical and Computer Engineering
University of California
Santa Barbara, CA 93106
Tel: (805) 893-4168, Fax: (805) 893-5705
Email: danb@ece.ucsb.edu.

1.0 Introduction

Internet traffic is growing exponentially and is forcing next generation Internet Protocol (IP) networks to scale far beyond present speed, capacity, performance, and packet forwarding rates. In fiber-based optical networks, next generation optical IP routing will require technologies to support packet routing and forwarding operations at Terabit wire rates that are compatible with wavelength division multiplexed (WDM) transmission and routing. Since up to 50% of IP traffic consists of packets smaller than 522 Bytes and 50% of these packets are in the 40-44 Byte range, new low-latency packet forwarding and routing technologies will be required that can handle wire-rate routing of the smallest packets in excess of Gigapackets/sec. These technologies should support new streamlined IP routing protocols such as Multiprotocol Label Swapping (MPLS) [1] that simplify route lookup and separate routing and forwarding functions.

All-Optical Label Swapping (AOLS) [2-4] is a method to implement MPLS optically and to couple the IP packet transport function directly to the WDM layer with flexible upgrade capabilities and support of Terabit wire rate routing. Implementation of AOLS in an optical core network is illustrated in Figure 1 where edge routers "encapsulate" IP packets by adding a subcarrier multiplexed (SCM) label and switch the packet to a new wavelength. The new wavelength is then used to perform part of the routing and forwarding functions within the core network. Within the network, Terabit optical core routers perform routing and forwarding functions, optical label swapping, and packet wavelength conversion.

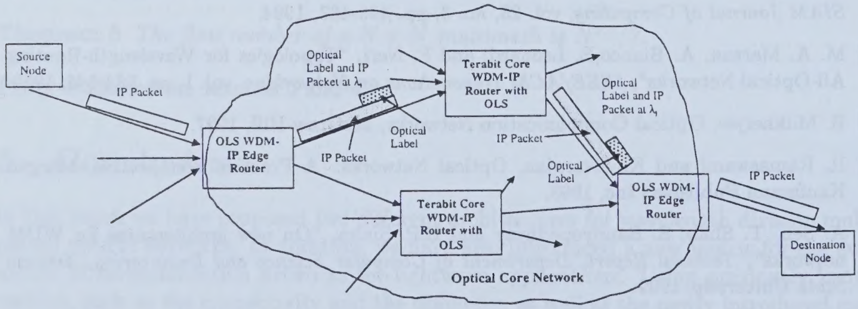

Source Node
IP Packet
Optical Label and IP Packet at λ_i
Terabit Core WDM-IP Router with OLS
Optical Label and IP Packet at λ_j
OLS WDM-IP Edge Router
Optical Label
IP Packet
IP Packet
Optical Label
IP Packet
OLS WDM-IP Edge Router
Terabit Core WDM-IP Router with OLS
Destination Node
Optical Core Network

Figure 1. Optical label Swapping with wavelength conversion in an optical core network using edge and core WDM-IP routers.

2.0 All-Optical Label Swapping (AOLS)

For future optical networks, it will be desirable to perform optical routing layer functions independent of the IP packet length and payload bit-rate. This leads to the concept of using optical labels to "encapsulate" the IP packet in the optical layer. The optical label format is chosen to best match the optical routing and forwarding technology and to satisfy other constraints in the optical layer (e.g., the current lack of flexible optical buffering technologies). Two primary approaches to optical label formatting are bit-serial and subcarrier multiplexed [5-7] as illustrated in Figure 2. A bit-serial label (a) is placed at the head of the IP packet following an optical guard-band (OGB). The OGB is use to facilitate label removal and reinsertion without static packet buffering and to accommodate finite switching times of optical switching and wavelength conversion. The bit-serial label is encoded on the same optical wavelength as the IP packet and is encoded as an electronic baseband signal. For optically subcarrier multiplexed (OSCM) labels (b) the optical label is first modulated onto a RF subcarrier and then modulated in parallel with the IP packet on the same wavelength. In both cases, the optical label bit-rate is chosen to balance the requirements for low cost label processing electronics and the duration of the label relative to the optical IP packet length. For current TCP/IP networks, the label duration should be chosen to facility high throughput routing of short 40 byte acknowledgement packets.

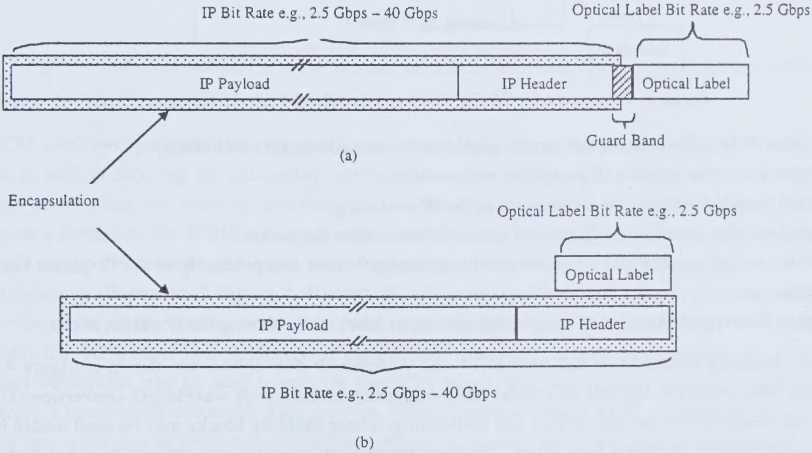

IP Bit Rate e.g., 2.5 Gbps – 40 Gbps
Optical Label Bit Rate e.g., 2.5 Gbps
IP Payload
IP Header
Optical Label
Guard Band
(a)
Encapsulation
Optical Label Bit Rate e.g., 2.5 Gbps
Optical Label
IP Payload
IP Header
IP Bit Rate e.g., 2.5 Gbps – 40 Gbps
(b)

Figure 2: Encapsulated optical IP packets using (a) bit serial and (b) subcarrier multiplexed optical labels.

3.0 AOLS with Optical Subcarrier Multiplexing

To a great degree, the success of WDM lies in the fact that for transmission between a source and destination, the endpoint electronics need to interface to only one wavelength rather than the aggregate fiber bandwidth. Optical subcarrier multiplexing (OSCM) is a technique to carry data on a secondary RF carrier modulated onto a primary optical carrier. Using OSCM, this concept can be extended to the network level by layering routing and control information on a lower bit-rate modulated subcarrier. This routing data can be recovered using low-cost electronics without accessing the payload bandwidth and IP packet. The process of generating SCM packets and recovering headers is described in [2-4,8,9]. An example transmitter and the resulting RF power spectrum for an OSCM encoded packet after photodetection is shown in Figure 3.

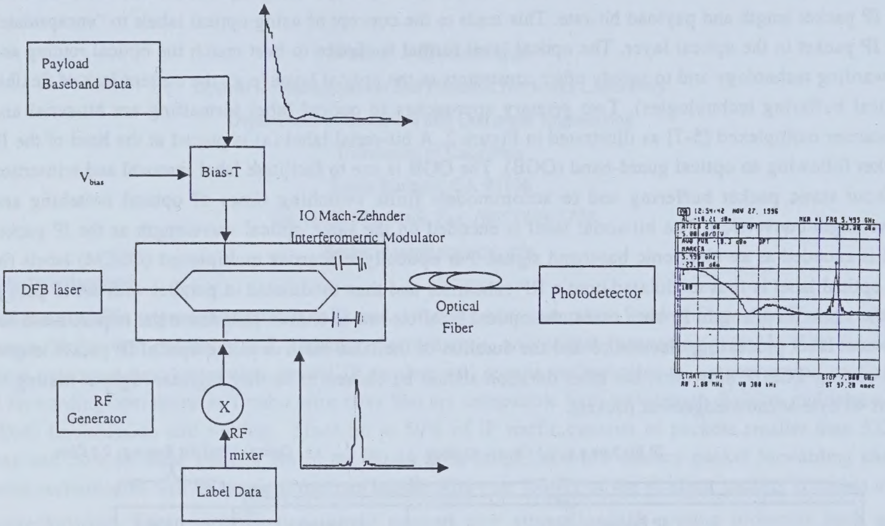

Payload
Baseband Data
V_{bias}
Bias-T
IO Mach-Zehnder
Interferometric Modulator
DFB laser
Optical
Fiber
Photodetector
RF
Generator
X
RF
mixer
Label Data
12:54:42 NOV 27, 1996
RL -10.21 dBm
MKR #1 FRQ 5.495 GHz
ATTEN 2 dB
-23.80 dBm
5.00 dB/DIV
AVG PWR -10.3 dBm
MARKER
5.495 GHz
-23.80 dBm
START 0 Hz
STOP 7.000 GHz
RB 3.00 MHz
VB 300 kHz
ST 97.20 msec

Figure 3: Transmitter for OSCM labels and detected RF power spectrum.

Encapsulation of IP packets using subcarrier labels has several advantages including:

- The contents of the original IP packet are not modified,
- the label is coded at the same wavelength as the IP packet,
- the label may be coded at a different bit-rate and format than the packet,
- detection, erasure and rewriting of the label may be performed independently of the IP packet bit-rate and phase and
- low pass filtering can be used to remove the subcarrier label while leaving the IP packet in tact.

The key building blocks to AOLS with SCM and wavelength conversion, are shown in Figure 4. A burst-mode label recovery tap and an optical label-swapping module with wavelength conversion (OLS-WC) enable header detection, reinsertion and forwarding. These building blocks may be used within both core and edge routers. Within a core router, the burst-mode label recovery tap extracts the label data and clock to process routing and forwarding functions electronically while passing an optical IP packet with SCM label to a 2-stage wavelength converter. The wavelength converter erases the old SCM label, converts the IP packet to a new wavelength, and rewrites a new SCM label onto the new wavelength. Within an edge router, the tap recovers the serially coded IP header from a packet, and the wavelength converter converts the IP packet to a new wavelength without modifying the contents and adds an SCM label.

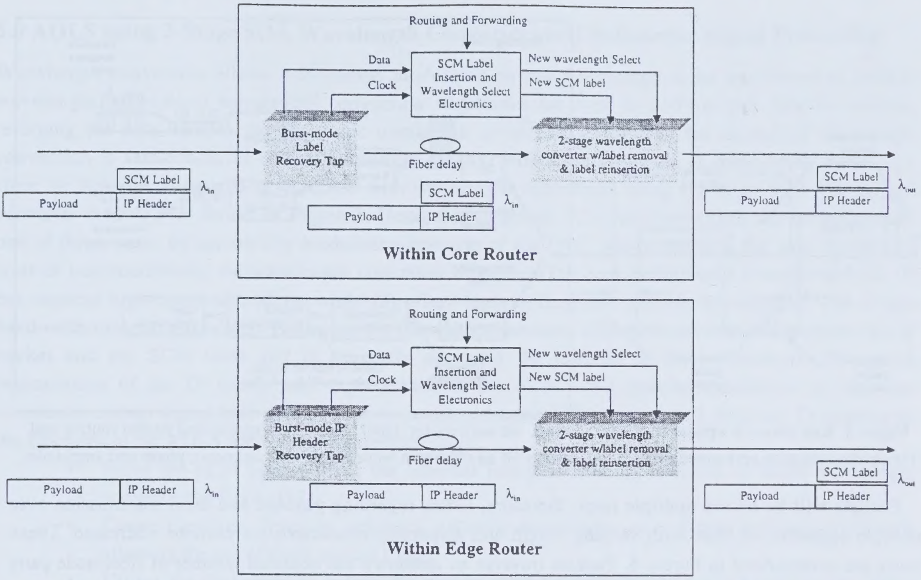

Routing and Forwarding
Data
Clock
SCM Label Insertion and Wavelength Select Electronics
New wavelength Select
New SCM label
Burst-mode Label Recovery Tap
Fiber delay
2-stage wavelength converter w/label removal & label reinsertion
SCM Label
Payload
IP Header
λ_{in}
λ_{out}
Within Core Router
Routing and Forwarding
Data
Clock
SCM Label Insertion and Wavelength Select Electronics
New wavelength Select
New SCM label
Burst-mode IP Header Recovery Tap
Fiber delay
2-stage wavelength converter w/label removal & label reinsertion
Payload
IP Header
λ_{in}
SCM Label
λ_{out}
Within Edge Router

Figure 4. OLS Core and edge IP router function using SCM labels and 2-stage wavelength conversion.

4.0 Issue in Multihop AOLS Networks

SCM label swapping separates the routing and forwarding functions into different electrical frequency bands as well as reducing the processing overhead at routers and providing independence to the payload bit-rate and format. By utilizing wavelength conversion to implement the label swapping process, we integrate a portion of the WDM routing function into the label-swapping module. At each router there are multiple issues that must be addressed when implementing an all-optical label swapping system based on SCM labels as illustrated in Figure 5. Packets of unknown amplitude and phase relative to a local time reference arrive at the routing node. A portion of the optical signal must be directed to a module that recovers the SCM label independently of the payload. Low-cost narrow-band electronics, blind to the payload bandwidth, can be used to recover the SCM label. The inter-packet spacing is also unknown, requiring a burst-mode subcarrier data receiver. When double sideband SCM modulation is used, carrier fading can occur due to fiber dispersion. The amount of carrier fading will vary depending on the inter-node distance and will help determine the required dynamic range of the burst-mode subcarrier receiver. Therefore, either signal strength must be normalized on a node by node basis (possible since the label is replaced at every node) or dispersion insensitive subcarrier recovery techniques must be used. The label envelope must be recovered from a detected subcarrier with unknown RF phase. Data and clock recovery must then be performed without knowledge of the optimal threshold or clock phase. The remaining power in the packet is directed to a label-swapping module. The old label must be removed independent of the payload bit-rate and format. For WDM-IP networks, wavelength conversion can facilitate packet forwarding. Payload digital regeneration is needed to support multihop routing. A new label must be replaced onto the outgoing optical packets with minimum knowledge of the payload contents.

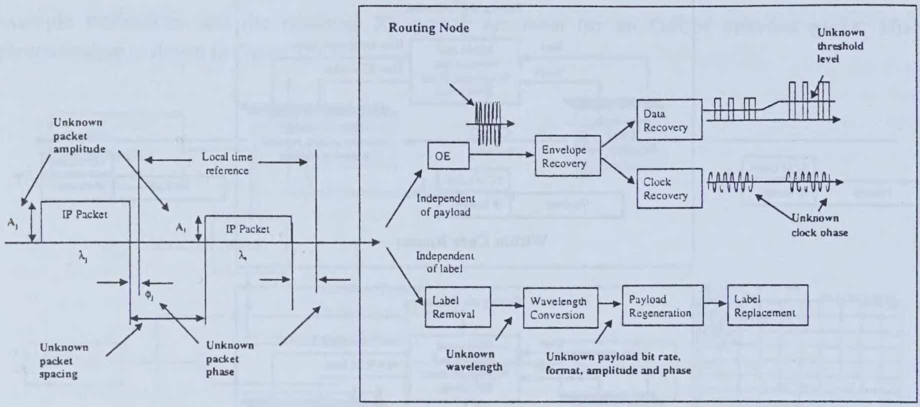

Routing Node
Unknown threshold level
Unknown packet amplitude
Local time reference
OE
Envelope Recovery
Data Recovery
Clock Recovery
Independent of payload
Unknown clock ohase
A_j
IP Packet
A_i
IP Packet
λ_j
λ_i
ϕ_j
Independent of label
Label Removal
Wavelength Conversion
Payload Regeneration
Label Replacement
Unknown packet spacing
Unknown packet phase
Unknown wavelength
Unknown payload bit rate, format, amplitude and phase

Figure 5. Key issues in optical IP packet routing. At each router, label recovery/swapping and packet routing and forwarding must be performed on data that arrives on an unknown wavelength with unknown phase and amplitude.

Packets will be routed multiple hops, therefore, issues regarding payload and label transmission over multiple segments of fiber with varying length and dispersion characteristics must be addressed. These issues are summarized in Figure 6. Packets traverse an unknown but bounded number of fiber/node pairs (hops) from source to destination. Each hop will be over a different fiber length with varying fiber dispersion. Three critical issues in maintaining payload and label signal integrity are: **(I)** Accumulated degradation of the payload SNR due to dispersion induced bit broadening, **(II)** accumulated noise and jitter from optical amplifiers and system nonlinearities, and **(III)** periodic fading of double-sideband (DSB) SCM labels due to fiber dispersion. These factors will degrade the payload and label, seriously over multiple hops, thereby limiting the scalability and performance of the IP network. Single sideband SCM signaling and suppressed carrier techniques will alleviate (III).

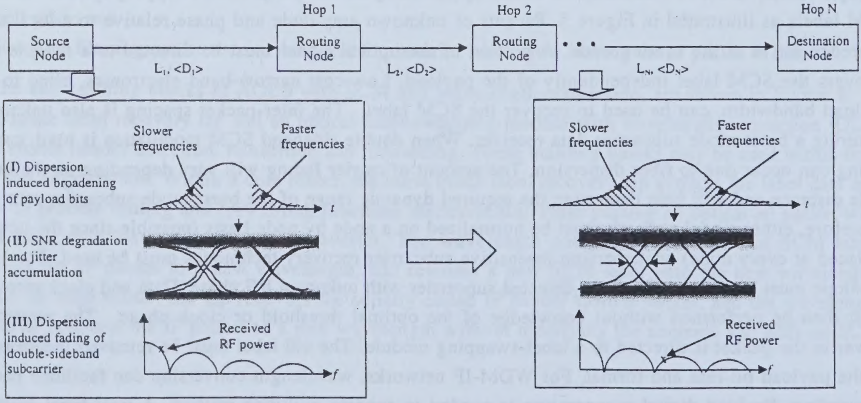

Hop 1
Hop 2
Hop N
Source Node
Routing Node
Routing Node
Destination Node
$L_1, <D_1>$
$L_2, <D_2>$
$L_N, <D_N>$
Slower frequencies
Faster frequencies
(I) Dispersion induced broadening of payload bits
(II) SNR degradation and jitter accumulation
(III) Dispersion induced fading of double-sideband subcarrier
Received RF power
t
f
Slower frequencies
Faster frequencies
Received RF power
t
f

Figure 6.Effect of (I) dispersion, (II) nonlinearities and (III) periodic fading of DSB-SC over multi hops

5.0 AOLS using 2-Stage SOA Wavelength Converters and Subcarrier Signal Processing

Wavelength conversion allows information modulated on one wavelength to be transferred to another wavelength. All-optical wavelength conversion techniques are used to perform this function without returning the data back to the electronic domain. A promising technology for all-optical wavelength conversion is semiconductor optical amplifiers (SOAs) which are discussed in more detail in [10,11]. Here we describe an approach to AOLS with wavelength conversion using a 2-stage SOA wavelength converter module [12] shown in Figure 7. Using this structure, SCM label rewriting can be achieved in one of three ways: by electrically modulating one arm of the IWC, pre-modulating the new wavelength laser or post-modulating the wavelength converted IP packet. The best performance is achieved with (1) but requires high-bandwidth SOAs while (2) offers high performance and can be achieved with a high bandwidth external modulator. Technique (3) introduces harmonic distortion and crosstalk between the IP packet and the SCM label and is generally not used. In addition to these functions, 2R optical regeneration of the IP packet and replacement of the SCM label enables cascadable routing and overcomes certain signal fading limitations to double sideband SCM transmission systems. To summarize the functions of the AOLS wavelength conversion stage:

- Erase the SCM header using the inherent low-pass filtering function of XGM wavelength conversion in SOAs [13]
- Convert packets from an arbitrary network wavelength to a fixed internal wavelength allowing the use of fixed optical filters
- Utilize the first stage to reduce the packet optical power dynamic range and set the packet average optical power level for second stage wavelength conversion
- Utilize an interferometric wavelength converter in the second stage to enhance the optical extinction ratio (2R regeneration) and convert the packet to an outgoing wavelength with a fixed filter
- Utilize a fast wavelength tunable laser to enable packet-rate wavelength conversion
- Remodulate the SCM label at the second converter stage

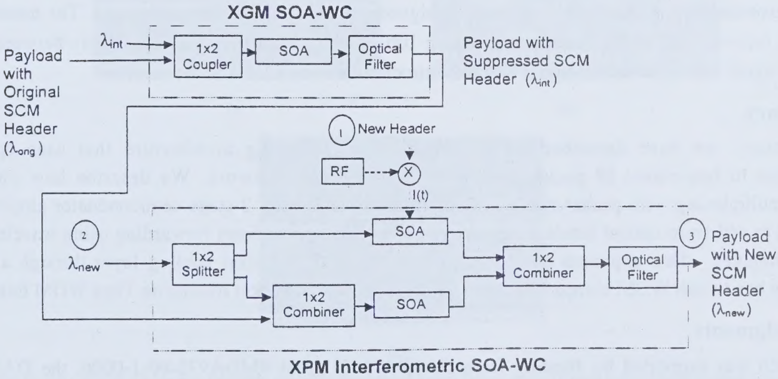

XGM SOA-WC
λint
Payload with Original SCM Header (λong)
1x2 Coupler
SOA
Optical Filter
Payload with Suppressed SCM Header (λint)
1
New Header
RF
X
I(t)
2
λnew
SOA
1x2 Splitter
1x2 Combiner
SOA
1x2 Combiner
Optical Filter
3
Payload with New SCM Header (λnew)
XPM Interferometric SOA-WC

Figure 7. A 2-stage SOA based wavelength converter for AOLS. Three methods to rewrite the optical label are indicated: (1) modulation of one arm of the IWC, (2) pre-modulation of the new laser wavelength or (3) post modulation of the wavelength converted IP packet.

6.0 Experimental Results

The WDM AOLS experiment is shown in Figure 8. An OSCM packet transmitter was used to generate a 150 Mbps label on an RZ coded, ASK modulated 16 GHz subcarrier. Labels consist of a 16-bit preamble, an 84-bit tag and a 4-bit framing sequence as shown in the upper trace of Figure 9a. The payload is an NRZ coded, 2.5 Gbps PRBS as shown in the lower trace of Figure 9a and the 16 GHz subcarrier supports payload bit rates out to 10 Gbps. The packet duration is 1 μs. Label clock and data are recovered on a packet-by-packet basis following a 10% fiber tap, an EDFA and an OSCM direct-detection receiver. The OSCM receiver utilizes a fast Schottky barrier diode for envelope detection. A SAW filter is used to recover the tag clock for each packet as shown in Figure 9b and a fixed digital delay is needed to realign data and clock. A tag switching processor is required to perform serial-to-parallel conversion, compute a new label, multiplex the new label onto a RF subcarrier and set a fast wavelength tunable laser to the new wavelength within 12 ns.

A two stage modified XGM/IWC is used to perform wavelength routing, optical label erasure and label rewriting at the IP packet rate. The OSCM label is removed during XGM due to the low pass frequency response of wavelength conversion in an SOA [5]. The XGM-WC converts incoming WDM packets to a fixed internal wavelength (λ_{int}) that is passed to the next stage using a fixed frequency optical filter. The XGM-WC is also used to set a stable operating point for the IWC. One arm of an InGaAsP IWC is injected with the optically filtered output of the XGM-WC. The output of a rapidly tunable 4-section GCSR laser transmitter is injected to both arms of the IWC.

Forwarding results for the WDM AOLS experiment are shown in Figure 9c. The packets are cycled among four wavelengths by tuning the GCSR LASER and individually detected using a tunable optical filter. The pedestal is due to incomplete extinction ratio sacrificed in order to cover the complete wavelength range of the tunable laser using the IWC. The final signal is non-inverted due to bit inversion of the XGM-WC and IWC. We measured the transmission BER for the label-switched payload with an observed maximum 4 dB power penalty for all four wavelengths as shown in Figure 10a. The power penalty is expected to decrease when optimally designed wavelength converters are used. The measured BER of the recovered SCM tag is shown in Figure 10b. The apparent increased sensitivity between the transmitted signal and tag switched signal is due to introduction of an EDFA at the receiver.

7.0 Summary

In summary, we have described an all-optical label swapping architecture that uses optical encapsulation to implement IP packet routing in a WDM core network. We describe how optical subcarrier multiplexing with packet-rate wavelength conversion using 2-stage semiconductor amplifiers can be used to add swap optical labels attached to an IP packet and perform forwarding using wavelength routing techniques. This approach can impact the future optical packet routing layer through a low latency, high bandwidth WDM compatible approach for wire-rate IP packet routing on Tbps WDM links.

Acknowledgments

This research was supported by funding from DARPA NGI grant #MDA972-99-1-0006, the DARPA sponsored Center on Multidisciplinary Optical Switching Technology (MOST) grant #F49620-96-1-0349 and a NSF National Young Investigator Award #ECS-9457148.

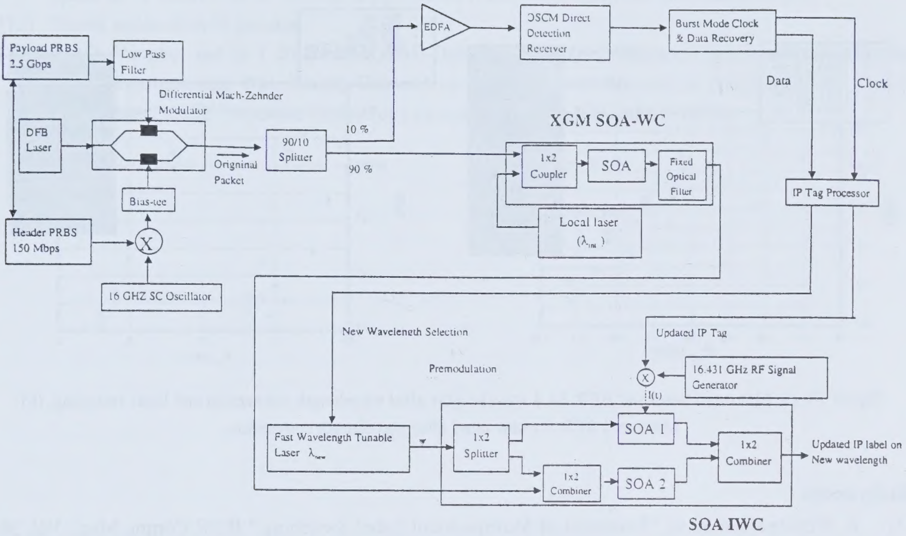
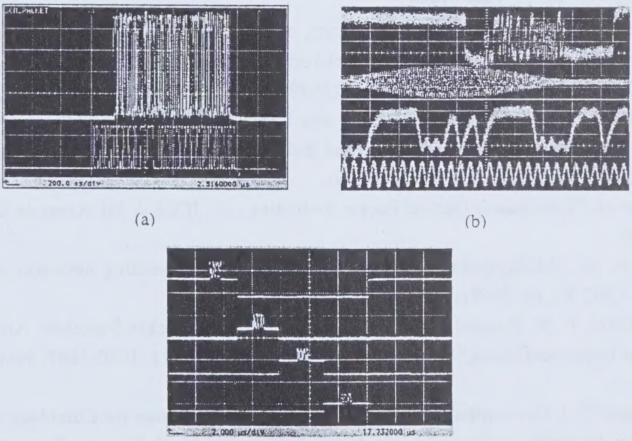

Payload PRBS
2.5 Gbps
Low Pass
Filter
Differential Mach-Zehnder
Modulator
DFB
Laser
Original
Packet
Bias-tee
Header PRBS
150 Mbps
16 GHZ SC Oscillator
90/10
Splitter
10 %
90 %
EDFA
OSCM Direct
Detection
Receiver
Burst Mode Clock
& Data Recovery
Data
Clock
XGM SOA-WC
1x2
Coupler
SOA
Fixed
Optical
Filter
Local laser
(λ_{int})
IP Tag Processor
New Wavelength Selection
Updated IP Tag
Premodulation
16.431 GHz RF Signal
Generator
l(t)
Fast Wavelength Tunable
Laser λ_{new}
1x2
Splitter
SOA 1
1x2
Combiner
SOA 2
1x2
Combiner
Updated IP label on
New wavelength
SOA IWC

Figure 8. Experimental setup for demonstrating AOLS using OSCM labels and SOA based wavelength conversion.



(a)

(b)

(c)

Figure 9: (a) OSCM IP packet (bottom trace) with optical label (top trace), (b) burst mode recovered label and (c) wavelength converted packets with new optical labels (each trace is at a different wavelength in the 1550 nm band).

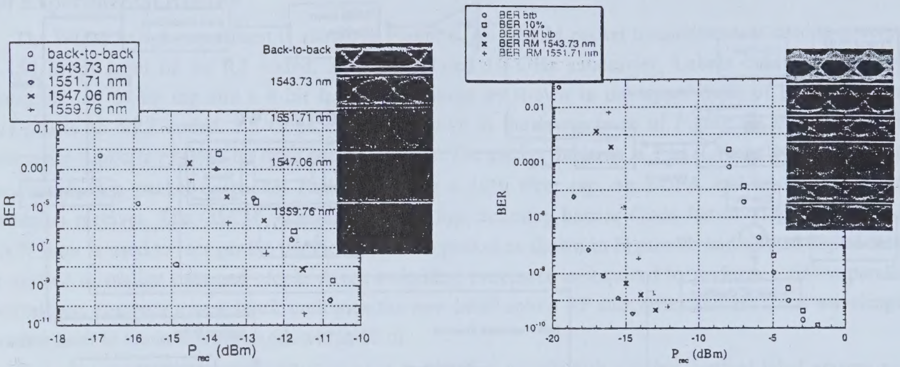


Figure 10: (a) Measured baseband BER for 4 wavelengths after wavelength conversion and label swapping. (b) Measured BER for new label after wavelength conversion.

References

[1] A. Viswanathan et. al, "Evolution of Multiprotocol Label Switching," IEEE Comm. Mag., Vol. 36, No. 5, pp. 165-173, May (1998).

[2] A. Carena, et. al. , "OPERA: An Optical Packet Experimental Routing Architecture with Label Swapping Capability," *IEEE Journal of Lightwave Technology, Special Issue on Photonic Packet Switching*, Vol. 16, No. 12, Dec. (1998).

[3] "D. J. Blumenthal, A. Carena, L. Rau, V. Curri, and S. Humphries, All-Optical Label Swapping with Wavelength Conversion for WDM-IP Networks with Subcarrier Multiplexed Addressing," To Appear in *IEEE Photonics Technology Letters* (1999).

[4] D. J. Blumenthal, A. Carena, L. Rau, V. Curri and S. Humphries, "WDM Optical IP Tag Switching with Packet-Rate Wavelength Conversion and Subcarrier Multiplexed Addressing," OFC/IOOC, Paper ThM1, San Diego, CA, Feb. 21-26 (1999).

[5] P. Gambini et al, "Transparent Optical Packet Switching:...", *IEEE J. Sel. Areas in Comm.*, vol. 16, pp.1245-1259.

[6] A. Budman et. al., "Multigigabit optical packet switch for self-routing networks with subcarrier addressing,", OFC '92, pp. 90-91, San Jose, CA, Feb. 4-7, (1992).

[7] D. J. Blumenthal, P. R. Prucnal, and J. R. Sauer, "Photonic Packet Switches: Architectures and Experimental Implementations," *Proceedings of the IEEE*, **82** (11), 1650-1667, November (1994). (Invited Paper).

[8] R. Gaudino and D. J. Blumenthal, "A Novel Transmitter Architecture for Combined Baseband Data and Subcarrier-Multiplexed Control Links Using Differential Mach-Zehnder External Modulators," *IEEE Photonics Technology Letters*, **9** (10), 1397-1399, October (1997).

[9] D. J. Blumenthal, J. Laskar, R. Gaudino, S. Han, M. Shell and M. D. Vaughn, "Fiber Optic Links Supporting Baseband Data and Subcarrier-Multiplexed Control Channels and the Impact of MMIC Photonic/Microwave Interfaces," *IEEE JLT/MTT Joint Special Issue on Microwave Photonics II, Transactions on Microwave Theory and Techniques*, **45**(8), 1443-1452, August 8 (1997).

[10] S. L. Danielsen et. al., "All Optical Wavelength Conversion Schemes for Increased Input Power Dynamic Range," *IEEE Photonics Technology Letters*, Vol. 10, pp. 60-62, Jan., (1998).

[11] B. Mikkelsen, et. al., "Transmission through 10 all-optical interferometric wavelength converter spans at 10 Gbit/s," OFC '96, Vol. 2, pp. 380-383, San Jose, CA, Feb. 25 – March (1996).

[12] Patent application in process.

[13] M. D. Vaughn, and D. J. Blumenthal, "All-Optical Updating of Subcarrier Encoded Packet Headers with Simultaneous Wavelength Conversion of Baseband Payload in Semiconductor Optical Amplifiers," *IEEE Photonics Technology Letters*, **9** (6), 827-829, June (1997).

On Discrete Sufficient Statistics for Acquisition in Asynchronous Band-limited CDMA Systems[1]

Ashok Mantravadi and Venugopal V. Veeravalli
School of Electrical Engineering
Cornell University
Ithaca, NY 14850
ashok, venu@ee.cornell.edu

Abstract

The derivation and analyses of delay estimation (acquisition) schemes in CDMA systems is sometimes simplified in the literature by relaxing the bandwidth constraints on the signaling waveforms. It is shown that this approach can produce inaccurate discrete system models leading to anomalous results in asynchronous CDMA systems. This is illustrated by considering discrete models generated using multiple correlators and chip-matched filtering. Maximum-likelihood acquisition schemes are derived based on decorrelating and MMSE strategies. The effect of relaxing bandwidth constraints is then studied through numerical simulations of the performance of these schemes.

I Introduction

The continuous-time system model that describes a typical Code Division Multiple Access (CDMA) system is one where the sum of the transmitted signal waveforms of the users goes through a (possibly time-varying) band-limited channel with additive background noise. It is convenient both for analysis as well as implementation of receiver operations to convert the continuous time CDMA model into an equivalent discrete model, which produces sufficient statistics for statistical decision making at the receiver.

In standard analyses of communication systems, the generation of sufficient statistics for detection/estimation at the receiver is facilitated by relaxing the bandwidth constraints on the continuous-time system model [11]. In particular, the background noise may be idealized to have infinite bandwidth. In addition, bandwidth constraints on the signaling waveforms may also be relaxed in cases where the performance metric of interest (e.g. bit-error probability) is independent of signal bandwidth. The advantage of relaxing bandwidth constraints in this fashion is that it allows for the use of time-limited (TL) pulses in analysis and exposition of operations at the receiver.

Much of the work on CDMA systems, especially that involving joint processing of the received signal (e.g. [2, 5]) has used the same approach of relaxing bandwidth constraints in the continuous-time model. Bandwidth considerations are usually ignored in the choice of the chip waveforms in the analyses, and waveforms time-limited to the chip period (such as the standard rectangular pulse) are used. In earlier work [6, 7], we evaluated the validity of this approach for detection and conventional acquisition in asynchronous CDMA systems. We now consider the same issue with respect to more general acquisition schemes at the CDMA receiver. We note that there are a few exceptions (e.g. [9]) which use band-limited (BL) pulse shapes while

---

[1] This research is supported in part by the NSF under grant NCR-9523967.

deriving acquisition schemes. We take a more general approach, and evaluate the validity of relaxing bandwidth constraints in the generation of discrete sufficient statistics for a large class of acquisition schemes.

We consider the acquisition problem on the asynchronous reverse link of the CDMA system, where the delay of a single (new) user needs to be estimated. If there is more than one new user in the system, they are assumed to be acquired in sequence. Since the remaining (interfering) users are already being demodulated, it seems reasonable to assume that their delays, and hence, the spreading sequences of the interferers are known. This setting is particularly useful for CDMA systems which employ long (random) spreading sequences. Three maximum-likelihood acquisition schemes are considered: the conventional acquisition scheme which treats the MAI as random noise; a decorrelating scheme which perfectly cancels the interfering users; and a minimum mean squared-error based scheme. These schemes are closely related to the the corresponding detection schemes: the basic idea is that we run the corresponding detector on a modified version of the incoming signal to get an estimate of the interferer's signal parameters (bits, phases etc.) and use these estimates to derive a likelihood function for the delay of the new user.

For the band-limited system, the above schemes can be derived based on the optimum discrete statistics generated by projecting the received signal onto a set of Prolate Spheroidal Wave functions [11]. However, such a scheme would be impractical, and we consider two related ways of deriving discrete statistics: chip-matched filtering (chip-MF) and correlation. The chip-MF approach involves projection onto a set of shifted chip waveforms, and has been used to discretize CDMA signals in many recent papers [5, 8]. The correlation-based schemes involve discrete statistics derived by projection onto the (known) spreading waveforms of the interferers. While the bandwidth restrictions on the noise can be relaxed under very general conditions, the issue of relaxing bandwidth constraints on the chip waveform needs to be considered more carefully. In this paper, the performance of the chip-MF and correlation-based schemes, with both time-limited (TL) and band-limited (BL) chip waveforms, are compared through simulations, and conclusions are drawn regarding the effect of relaxing bandwidth constraints on acquisition.

II System model

We consider a BPSK DS/CDMA model with K users, where the received complex baseband signal is given by

$$y(t) = \sum_{n=-1}^{M-1} \sum_{k=1}^{K} \sqrt{2\mathcal{E}_k}\, b_{k,n} c_k^{(n)}(t - \tau_k) e^{j\phi_k} + w(t), \quad t \in [0, MT_s], \tag{1}$$

and where

- $b_{k,n}$ is symbol n of user k, and T_s is the symbol period.
- $c_k^{(n)}(t) = \sum_{j=0}^{N-1} c_{k,j}^{(n)}\, \psi(t - nT_s - jT_c)$. Here T_c is the chip period, $\psi(t)$ is the unit-energy chip waveform, $c_{k,j}^{(n)}$ is the $\pm\frac{1}{\sqrt{N}}$ chip sequence with and $N = T_s/T_c$ is the processing gain of the system.

- ϕ_k, τ_k and $\mathcal{E}_k$ are, respectively, the carrier phase offset, delay and the received symbol energy of the user k. Define $A_k = \sqrt{2\mathcal{E}_k}$ to be the corresponding received amplitude.
- $w(t)$ is a zero mean circularly complex Gaussian process with two-sided power spectral density N_0, i.e. $R_w(\tau) = \frac{1}{2}E[w^*(t)w(t+\tau)] = N_0\delta(\tau)$.

Thus we are considering an M-symbol observation of the received signal. In a synchronous situation, the symbols would be indexed from $n = 0$ to $n = M - 1$. However, since the users are asynchronous, there is an additional symbol corresponding to $n = -1$ in the observation window.

The new user entering the system is taken to be the one with $k = 1$. In addition, we assume the following:

- The initial timing uncertainty is L chips, i.e.,$\tau_1 \in [0, LT_c)$. Then, we can write $\tau_1/T_c = n_1+\alpha_1$, with $n_1 \in \{0, 1 \ldots L-1\}$ and $\alpha_1 \in [0, 1)$. And for $k \neq 1, \tau_k = (n_k+\alpha_k)T_c \in [0, T_s)$.
- The new user transmits a preamble with the all one sequence over the observation interval: $b_{1,n} = 1, \forall n$.
- We have coherent demodulation and perfect carrier synchronization with respect to user 1, so that $\phi_1 = 0$.
- The users undergo slow flat fading, and the symbol energy in (1) includes the fading effect.

The last two assumptions are made for simplicity. Consequently, the received signal of interest reduces to

$$y(t) = A_1 c_1(t - \tau_1) + \sum_{n=-1}^{M-1} \sum_{k=2}^{K} A_k b_{k,n} c_k^{(n)}(t - \tau_k) \cos \phi_k + n(t), \tag{2}$$

where $n(t) = \mathrm{Re}\{w(t)\}$ is a real Gaussian process with two sided PSD $N_0/2$.

Since user 1 employs a preamble, this model can be converted to a problem involving a maximum of $K_{\text{eff}} = (K-1)(M+1) + 1$ effective users by separating the signals corresponding to each bit of each user occurring in the observation interval (see Fig. 1)[2]:

$$y(t) = A_1 \tilde{c}_1(t) + \sum_{k=2}^{K_{\text{eff}}} a_k \tilde{c}_k(t) + n(t), \tag{3}$$

where $\tilde{c}_k(t)$ is the spreading waveform corresponding to the k^{th} effective user. The bits of the interfering users are not of immediate interest, and they are grouped with the amplitudes and phases as

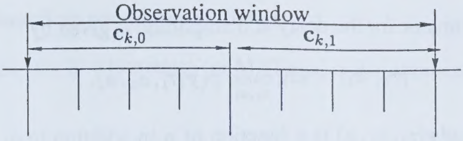
$$a_{(M+1)(k-2)+n+1} = A_k b_{k,i} \cos \phi_k, \quad k = 2, \ldots, K_{\text{eff}} \text{ and } n = 1 \ldots M+1. \tag{4}$$

Now, the generation of discrete sufficient statistics from the continuous-time signal $y(t)$ involves the projection of $y(t)$ onto a set of functions $\{g_n(t)\}$[10, 7]. Then, we have

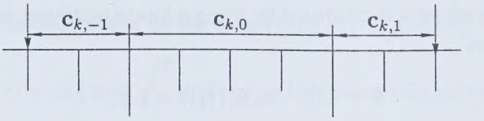
$$y_n = \int_0^T y(t) g_n(t) dt = < y(t), g_n(t) >$$

---

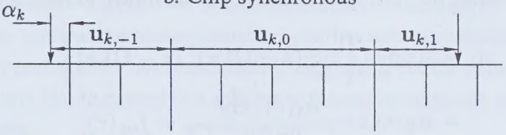
[2] The band-limited chip waveforms would of course extend beyond the chip 'boundaries' shown in Fig. 1.



a. Symbol synchronous



b. Chip synchronous



c. Chip asynchronous

Figure 1: Illustration of model with effective users: $M = 2$

---

for $n = 1 \ldots D$, where D could represent a dimensional restriction in the model. Since this is a linear operation, the resulting vector $\mathbf{y} = [y_1, y_2, \ldots, y_D]^\top$ can be represented as

$$\mathbf{y} = a_1 \mathbf{u}_1(\tau_1) + \mathbf{U}\mathbf{a} + \mathbf{n}, \tag{5}$$

where $u_{ij} =< g_i(t), \tilde{c}_j(t) >$, $\mathbf{U} = [\mathbf{u}_2, \ldots, \mathbf{u}_{K_{\text{eff}}}]$ is a matrix of *only* the interfering user vectors, $\mathbf{a} = [a_2 \ldots a_{K_{\text{eff}}}]^\top$, and $\mathbf{n}$ is a real Gaussian vector with $n_i =< g_i(t), n(t) >$.

Since we have a time-limited observation of a band-limited signal, the optimum basis set for discretization is the set of the Prolate Spheroidal Wave functions (PSWFs) [4]. However, the resulting signal model would be impractical for analysis and implementation. Hence, we consider two simpler cases for the set of functions $\{g_n\}$. Since we have white Gaussian noise, the optimum scheme based on the continuous time signal can be derived using correlations of the received signal with the spreading waveforms of the users, $g_n(t) = \tilde{c}_n(t)$, $n = 1, \ldots, K_{\text{eff}}$ (see section III.1). We refer to this as the correlator approach. The second approach, Chip-matched filtering (chip-MF), corresponds to using translated chip waveforms $\{\psi(t - nT_c)\}$ ($n = 1, \ldots, MN$) for discretization. The discrete system model using chip-matched filtering can be formed without the knowledge of the delays of all the users, and hence, it is particularly useful for solving the acquisition problem. Also, chip-MF is optimum asymptotically as the observation interval increases [7]. The equivalence of the two approaches, in relation to the bandwidth constraints imposed on the system, is of interest.

III Acquisition Schemes

We begin by considering the discrete system model (5), and assume that chip-matched filtering is used for the discretization, since it makes the exposition of the schemes simpler. The maximum likelihood estimator for the delay and amplitude is given by

$$[\hat{\tau}_1,\ \hat{a}_1] = \arg\max_{\tau_1, a_1}\ p(\mathbf{y}|\tau_1, a_1, \mathbf{a}). \tag{6}$$

The likelihood function $p(\mathbf{y}|\tau_1, a_1, \mathbf{a})$ is a function of $\mathbf{a}$ in addition to a_1 and τ_1. Define $\mathbf{z} = (\mathbf{y} - a_1\mathbf{u}_1(\tau_1))$ as the effective received vector without the new user. We consider a class of acquisition algorithms where $\mathbf{a}$ is estimated by using a *linear* multiuser detection scheme, i.e., for fixed a_1, τ_1, we have

$$\hat{\mathbf{a}} = \mathbf{L}(\mathbf{y} - a_1\mathbf{u}_1(\tau_1)) = \mathbf{L}\mathbf{z}. \tag{7}$$

Since the noise is Gaussian, the likelihood solution can be shown to reduce to

$$\begin{aligned} \hat{\tau}_1 &= \arg\min_{\tau_1}\left\{\min_{a_1}(\mathbf{z} - \mathbf{U}\mathbf{L}\mathbf{z})^\top(\mathbf{z} - \mathbf{U}\mathbf{L}\mathbf{z})\right\} \\ &= \arg\max_{\tau}\frac{\mathbf{u}_1(\tau)^\top\mathbf{B}\mathbf{y}}{\sqrt{\mathbf{u}_1(\tau)^\top\mathbf{B}\mathbf{u}_1(\tau)}} = f_{\mathrm{ML}}(\tau), \end{aligned} \tag{8}$$

where $\mathbf{B} = (\mathbf{I} - \mathbf{U}\mathbf{L})^\top(\mathbf{I} - \mathbf{U}\mathbf{L})$. The matrix $\mathbf{L}$ could be chosen to satisfy any suitable criterion, depending on the information available regarding the interferers and the computational resources at the receiver. We consider three special cases. The conventional acquisition scheme treats the MAI as random noise and does not estimate $\mathbf{a}$, so $\mathbf{L} = \mathbf{0}$. For a decorrelating strategy, $\mathbf{L} = (\mathbf{U}^\top\mathbf{U})^{-1}\mathbf{U}^\top$. And when we use a MMSE criterion for estimating $\mathbf{a}$, $\mathbf{L} = \mathbf{A}(\mathbf{U}^\top\mathbf{U} + \sigma^2\mathbf{A}^{-2})^{-1}\mathbf{U}^\top$, where $\mathbf{A}$ is a diagonal matrix consisting of the amplitudes and phase terms of the interfering users: $A_{ii} = A_k\cos\phi_k$, with the i^{th} effective interferer being associated with the k^{th} user.

We still need to maximize the statistic in (8) over τ to get an estimate of the delay. This maximization can be done analytically when the chip waveform is time-limited to $[0, T_c]$. In this case, we have

$$\mathbf{u}_k(\tau_k) = R_\psi(\alpha_k)\tilde{\mathbf{c}}_k(n_k) + R_\psi(1 - \alpha_k)\tilde{\mathbf{c}}_k(n_k + 1), \tag{9}$$

where $\tau_k = (n_k + \alpha_k)T_c$, $R_\psi(\alpha_k)$ is the auto-correlation of the chip-waveform at $\alpha_k T_c$, and $\tilde{\mathbf{c}}_k(n_k)$ is the vector corresponding to the sequence of k^{th} effective user. The algorithm for the optimization (8) consists of evaluating $f_{\mathrm{ML}}(\tau)$ at nT_c, $n \in \{0, 1 \ldots L-1\}$ and finding the local maxima in each of the intervals $(nT_c, (n+1)T_c]$ (see, e.g., [1]). A global maximum is then taken over all the values obtained and the corresponding τ is the ML estimate. Using (9), the maximum of $f_{\mathrm{ML}}(\tau)$ in $(nT_c, (n+1)T_c]$ occurs at $nT_c + \alpha_m(n)T_c$, with $\alpha_m(n)$ given by

$$\frac{R_\psi(\alpha_m)}{R_\psi(1 - \alpha_m)} = \frac{p_n q_{n+1} - p_{n+1} r_n}{p_{n+1} q_n - p_n r_n}. \tag{10}$$

where

$$p_n = \tilde{\mathbf{c}}_1(n)^\top\mathbf{B}\mathbf{y},\ q_n = \tilde{\mathbf{c}}_1(n)^\top\mathbf{B}\tilde{\mathbf{c}}_1(n)\ \text{ and }\ r_n = \tilde{\mathbf{c}}_1(n)^\top\mathbf{B}\tilde{\mathbf{c}}_1(n+1)\,. \tag{11}$$

In particular, for the rectangular pulse, $R_\psi(\alpha) = 1 - \alpha$, and we have

$$\alpha_m(n) = \frac{p_{n+1}q_n - p_n r_n}{p_n q_{n+1} + p_{n+1}q_n - r_n(p_n + p_{n+1})}\,. \tag{12}$$

III.1 Acquisition based on correlator statistics

When we have a single user in white noise (or MAI modeled as white noise) the received signal in (2) reduces to

$$y(t) = A_1 c_1(t - \tau_1) + n(t).$$

The ML estimate of the delay based on the observation $y(t)$ is given by

$$\hat{\tau}_1 = \arg\max_{\tau} \int_0^T c_1(t - \tau) y(t) dt = \arg\max y_1(\tau) \tag{13}$$

where we assumed $< c_1(t-\tau), c_1(t-\tau) >$ is independent of τ, and $y_1(\tau) =< c_1(t-\tau), y(t) >$. This is of course the basis of the conventional acquisition scheme, which involves generating a discrete statistic for each fixed τ, and maximizing the statistic for the values of τ in a continuous range. We would now like to extend this scheme to the multi-user case using the decorrelating and MMSE strategies.

Now, the solution based on the chip-MF statistics was closely related to the linear multiuser detection. We could follow a similar approach by forming the effective received signal $z(t) = y(t) - a_1 c_1(t - \tau)$ for fixed a_1 and τ. Since we assume that the spreading waveforms of the interferers are known, the correlation statistics can be formed as

$$< \tilde{c}_k(t), z(t) >=< \tilde{c}_k(t), y(t) - a_1 c_1(t - \tau)) >= y_{\mathrm{I}}(k - 1) - a_1 \rho(k - 1), \quad k = 2, \ldots, K_{\text{eff}}.$$

The estimate of $\mathbf{a}$ can written analogous to (7) as

$$\hat{\mathbf{a}} = \mathbf{L}_c(\mathbf{y}_{\mathrm{I}} - a_1 \boldsymbol{\rho}(\tau)), \tag{14}$$

where $\mathbf{y}_{\mathrm{I}} = [y_{\mathrm{I}}(1), \ldots, y_{\mathrm{I}}(K_{\text{eff}} - 1)]^\top$ and $\boldsymbol{\rho}(\tau) = [\rho(1), \ldots, \rho(K_{\text{eff}} - 1)]^\top$.

Instead of going through the details, we can arrive at the correlator solution quickly by noting that the inner product $\mathbf{x}^\top \mathbf{y}$ in the chip-MF case can be replaced by the inner product $< x(t), y(t) >= \int_0^T x(t) y(t)\, dt$ in the correlator case. In particular, the statistic $(\mathbf{y}_{\mathrm{I}} - a_1 \boldsymbol{\rho})$ is analogous to $\mathbf{U}^\top \mathbf{z} = \mathbf{U}^\top \mathbf{y} - a_1 \mathbf{U}^\top \mathbf{u}_1(\tau)$. Hence, comparing (7) and (14), we can draw analogies as $\mathbf{U}^\top \mathbf{y} \sim \mathbf{y}_{\mathrm{I}}$, $\mathbf{u}_1(\tau)^\top \mathbf{y} \sim \boldsymbol{\rho}$ and $\mathbf{L} \sim \mathbf{L}_c \mathbf{U}^\top$. Consequently, noting that $\mathbf{B} \sim (\mathbf{I} - \mathbf{U}\mathbf{L}_c\mathbf{U}^\top)^\top (\mathbf{I} - \mathbf{U}\mathbf{L}_c\mathbf{U}^\top)$ the chip-MF solution (8) can be shown to be analogous to

$$f_{\mathrm{ML}}(\tau) = \frac{\mathbf{u}_1(\tau)^\top \mathbf{B} \mathbf{y}}{[\mathbf{u}_1(\tau)^\top \mathbf{B} \mathbf{u}_1(\tau)]^{1/2}} = \frac{y_1(\tau) - \boldsymbol{\rho}(\tau)^\top \mathbf{B}_c \mathbf{y}_{\mathrm{I}}}{[\rho_{11} - \boldsymbol{\rho}(\tau)^\top \mathbf{B}_c \boldsymbol{\rho}(\tau)]^{1/2}}, \tag{15}$$

where $\rho_{11} =< c_1(t - \tau), c_1(t - \tau) >$, $\mathbf{B}_c = 2\mathbf{L}_c^\top - \mathbf{L}_c^\top \mathbf{R}_{\mathrm{I}} \mathbf{L}_c$ and $\mathbf{R}_{\mathrm{I}}$ denotes the correlation matrix of the spreading waveforms of the effective interferers. In particular, $\mathbf{L}_c = \mathbf{R}_{\mathrm{I}}^{-1}$ with the decorrelating strategy, and $\mathbf{L}_c = \mathbf{A}(\mathbf{R}_{\mathrm{I}} + \sigma^2 \mathbf{A}^{-2})^{-1}$ under the MMSE criterion. Note that the correlator statistics can be written as

$$\begin{pmatrix} y_1(\tau) \\ \mathbf{y}_{\mathrm{I}} \end{pmatrix} = \begin{pmatrix} \rho_{11} & \boldsymbol{\rho}(\tau)^\top \\ \boldsymbol{\rho}(\tau) & \mathbf{R}_{\mathrm{I}} \end{pmatrix} \begin{pmatrix} a_1 \\ \mathbf{a} \end{pmatrix} + \mathbf{n}, \tag{16}$$

and the noise vector $\mathbf{n}$ is not white in this case.

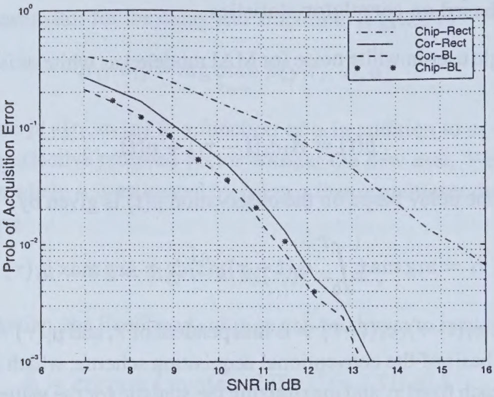


Figure 2: Single user acquisition performance, N=31, L=31.

The above correlator-based schemes are not attractive from an implementation perspective, since the correlation vector ρ needs to be computed for each value of the delay τ. However, the correlator scheme is optimum for the received continuous time signal in white noise, under the assumptions we made in deriving the ML estimator. Hence, it serves to illustrate the importance of discretization and the bandwidth restriction for the acquisition problem.

IV Numerical results

The performance of the above schemes are compared with BL and TL chip waveforms. Where an analytical simplification is not possible, a brute force maximization was performed with respect to τ_1. The performance metric used is the acquisition error probability, where acquisition failure occurs when $\hat{\tau}_1$ is more than 0.5 chips from τ_1. We are interested in the equivalence of the chip-MF and the correlator approaches in the band-limited case, and whether the performance results obtained by relaxing bandwidth constraints in the system are representative of true performance. Throughout the simulations, rectangular waveforms are used for the TL case, while a sinc pulse truncated to a length of 9 chips is used in the band-limited case. Also, the observation interval is taken to be just one symbol with $N = 31$, and the uncertainty interval is set to $L = 31$ chips as well.

In general, when the bandwidth restriction is imposed, the correlating scheme is both optimum and representative, while the chip MF can be expected to be close to the correlator performance [7]. This is clearly seen in Fig. 2, where the single user acquisition schemes based on the chip-MF and correlator perform identically when BL waveforms are used. On the other hand, there is a significant difference between the curves when TL rectangular pulses are used. The chip-MF scheme with TL waveforms is significantly off from the actual acquisition performance.

The degradation in performance of the chip-MF can be understood as follows. When rectangular pulses are used, the chip-MF statistics are not sufficient. In particular, the received signal is not spanned by $\{\psi(t - nT_c)\}$ for non-integral values of τ_1 and there is a loss in the energy

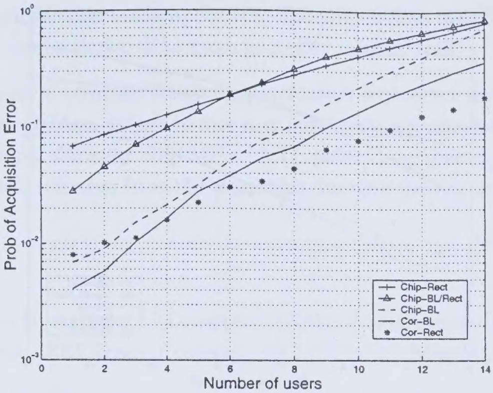


Figure 3: Performance of decorrelating acquisition scheme: $N = 31,\ M = 1$, SNR = 12 dB.

captured by these functions. We note that, as pointed out in [5], the energy loss due to chip-matched filtering can be recovered by oversampling. However, the resulting performance can at best correspond to that of the correlator with rectangular pulses, and the difference from the performance of the band-limited system will remain non-negligible.

When the bandwidth restriction is not imposed, the correlating scheme does not give representative results since it does not impose any dimensional restriction on the system. This is illustrated in Fig. 3, which shows the performance variation of the decorrelating acquisition scheme with the number of users K. The window length is one symbol which leads to rather high values of P_e, especially for large K. But it is the relative behavior of the schemes which is of interest. The dimensional issue becomes more important as the number of users increases (for fixed N) and the correlator with rectangular pulses gives overly optimistic results at higher values of K. Thus, designs and analyses of joint acquisition schemes which start with correlator statistics, but go on to use chip waveforms time-limited to $[0, T_c]$, may not be representative or useful in practice. The chip-MF with rectangular pulses continues to suffer from the loss in energy captured.

Note that the chip-MF with BL waveforms performs significantly worse than the corresponding correlator based scheme as K increases. In addition to the fact that, unlike the correlator, the chip-MF imposed a hard limit on the number of available dimensions, this sensitivity to K arises because we are using a decorrelating acquisition scheme. The decorrelating multi-user *detector* is known (see, e.g. [3]) to be quite sensitive to dimensional effects and, in fact, does not exist when $(M+1)K > MN$. In contrast, the MMSE-based scheme does not suffer from this sensitivity problem. The corresponding performance curves are shown in Fig. 4, where it is seen that the performance of the chip-MF and correlator schemes with BL waveforms remain close for all values of K.

Finally, since rectangular chip pulses do indeed facilitate the derivation of the ML delay estimate (see (10)), it is of interest to see how the ML estimate based on chip-matched filtering

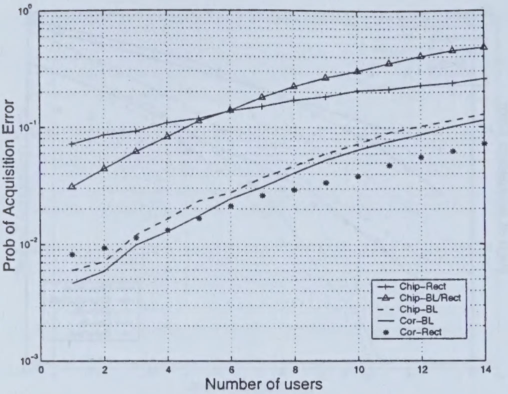


Figure 4: Performance of MMSE acquisition scheme.

---

with rectangular pulses performs when applied to the band-limited received signal. The resulting performance is shown as the 'Chip-BL/Rect' curve in Fig. 3 and Fig. 4. For a small number of users, the rectangular chip-MF captures more energy and performs better when the signal is band-limited than when the signal has rectangular pulses. However, as the number of users increases, the effect of the model mismatch becomes more severe. In any case, it is clear that the performance is much worse than the true performance of the band-limited system.

V Conclusions

The role of the bandwidth restrictions in generating discrete sufficient statistics for acquisition in CDMA systems was studied. Two means for discretization, chip-matched filtering and correlation with spreading waveforms were considered. In each case, maximum-likelihood acquisition schemes for a single new user were derived based on implicit linear multi-user detection for the interferers. Specifically, the conventional acquisition scheme and ML schemes based on the decorrelating and MMSE detectors were considered.

The effect of the relaxing the bandwidth restriction on acquisition performance was studied and the behavior of the correlator and chip-MF based schemes were compared. It was found that, in general asynchronous systems, using chip pulses time-limited to the chip interval may produce results which are not representative of performance in actual band-limited systems. Specifically, the results indicate that two different issues involved are: the signal energy captured and the dimensional restriction imposed on the system. When TL pulses are used, the chip-MF fails to capture signal energy and performs poorly. More importantly, the closed form solutions based on chip-matched filtering with TL pulses cannot be applied to the actual band-limited system due to the significant loss in performance. The correlator with TL pulses, on the other hand, does not impose any dimensional restriction on the system and gives overly optimistic results. This dimensional effect is seen more clearly for the decorrelation and MMSE based schemes. The correlator and chip-MF performances narrow down to the true performance of the system when the bandwidth restrictions are imposed. Note that, though several simplifying assumptions were made in deriving and simulations the acquisition schemes, the above observations can be shown to be more generally valid as well.

Finally, it is of interest to compare the above conclusions with those for the detection problem [7]. The acquisition problem differs in that it is inherently asynchronous with respect to the user of interest. Hence, while the chip-MF with TL pulses may give representative results for the detection problem, it is seen from this paper that the same cannot be said of the acquisition problem.

References

[1] S. Bensley and B. Aazhang. Maximum-likelihood synchronization of a single user for CDMA systems. *IEEE Trans. Commun.*, 46(3):392–399, March 1998.

[2] D. Zheng et al. An efficient code-timing estimator for DS-CDMA signals. *IEEE Trans. Signal Proc.*, 45(1):82–89, January 1997.

[3] Kiran and D. Tse. Effective bandwidth and effective interference for linear multiusers in asynchronous channels. *Submitted to IEEE Trans. Inform. Th.*, December 1998.

[4] H. J. Landau and H. O. Pollack. PSWFs-III: The dimension of the space of essentially time- and band-limted signals. *Bell Syst. Tech. J.*, pages 1295–1320, July 1962.

[5] U. Madhow. Blind adaptive interference suppression for direct-sequence cdma. *Proc. IEEE*, 86(10):2049–2069, Oct 1998.

[6] A. Mantravadi and V. Veeravalli. On discrete sufficient statistics for detection in asynchronous band-limited CDMA systems. In *Proc. 1999 CISS*, Baltimore, MD, March 1999.

[7] A. Mantravadi and V.V. Veeravalli. On discrete sufficient statistics for asynchronous band-limited CDMA systems. *Sumbitted to IEEE Trans. on Commun.*, Dec 1998.

[8] T. Ostman and B. Ottersten. Low complexity asynchronous DS-CDMA detectors. In *Proc. IEEE 46th Vehic. Technol Conf.*, pages 559–563, Atlanta, GA, May 1996.

[9] T. Ostman and B. Ottersten. Near far robust time delay estimation for asynchronous DS-CDMA systems with bandlimited pulse shapes. In *Proc. IEEE 48th Vehic. Technol Conf.*, pages 1650–1654, Ottawa, Canada, May 1998.

[10] H. Vincent Poor. *An Introduction to Signal Detection and Estimation*. Springer-Verlag, New York, 1994.

[11] J. Wozencraft and I. Jacobs. *Principle of Communication Engineering*. John Wiley and Sons, New York, 1965.

Frequency Synchronization Algorithm for Frequency Hopping System based on Singular Value Decomposition

A. Pouttu
University of Oulu, Telecommunication Laboratory, P.O. Box 4500,
90401 Oulu, Finland Tel: +358-8-5532827, Fax: +358-8-5532845, Email:
ari.pouttu@ee.oulu.fi

Abstract

In this paper baseband carrier frequency error estimators suitable for slow frequency hopping (SFH) system employing binary differential PSK (DPSK) as the data modulation are considered. Two cases are examined: pilot channel assisted estimation and extraction of frequency error from the data samples. In systems where the error changes slowly, every M^{th} frequency hop could be used as a pilot channel for frequency error estimation. In systems where the error needs to be constantly tracked, the error needs to be extracted from the data stream on hop-to-hop basis. A novel singular value decomposition (SVD) based method is introduced and the performance is compared to averaging filtering and absolute average filtering estimators.

Keywords

Frequency Synchronization, SVD, Frequency Hopping.

I. INTRODUCTION

In a frequency hopping system the center frequency of the carrier is pseudo randomly picked out of a large pool of frequencies. Each center frequency is maintained only for a brief interval T_f , i.e. the duration of a frequency hop. If the locally generated beat frequencies in the receiver aren't perfectly on the same frequency as the received signal, performance degradation will be introduced. The performance degradation is a function of the difference in frequency between the signals, the frequency error, and hence the means to minimize the difference are of importance. In a frequency hopping system frequency error can be introduced, apart from the synthesizer inaccuracies which are generated by the insufficient settling time of frequency synthesizers, by the Doppler effect generated by the movement of the transmitter or receiver. The instantaneous Doppler frequency is a function of the carrier frequency and thus in a frequency hopping

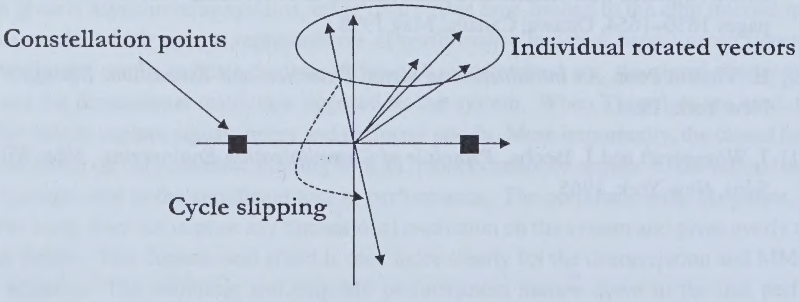


Fig. 1. Representation of the cycle slipping or equivocation problem.

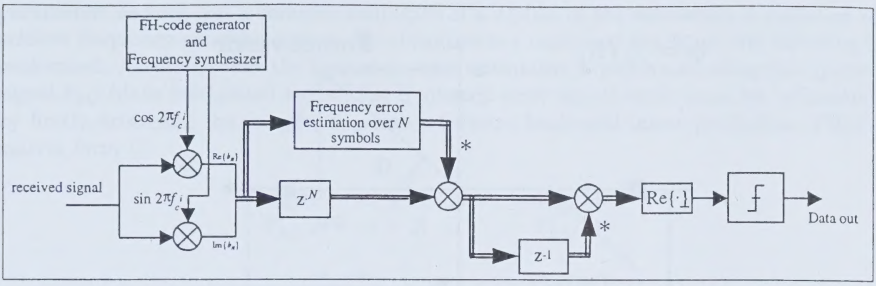


Fig. 2. The studied receiver structure with frequency error synchronization algorithm.

system a random variable. Hence frequency trackers are more suitable in cases where the number of the transmitted bits in a frequency hop is sufficiently high. In cases where the number of the transmitted bits in a frequency hop is small, a feed forward type estimator is a more practical approach. In this paper the assumption is made that the frequency errors are generated by the Doppler effect and frequency synchronizer inaccuracies and hence the knowledge of the frequency error during the previous frequency hop is not (cannot be) exploited. The problem associated with the frequency error estimation in a FH-system is the limited number of data samples / frequency hop. Hence a problem known as the equivocation [5] or cycle slipping [6] is introduced (see Fig. 1). A number of individual angle estimates obtained suffer from half plane ambiguity and will produce a biased estimate for instance if averaging filtering is used to filter the individual estimates. The aim of this paper is to present estimators that exhibit resistance to the equivocation phenomenon. The emphasis is on baseband automatic frequency correction (AFC) algorithms hence restricting to normalized frequency errors of $[-\frac{1}{2},\frac{1}{2}]$ for pilot channel estimation and $[-\frac{1}{4},\frac{1}{4}]$ when estimating from the data modulated signal.

II. SYSTEM MODEL

In a DPSK modulated system, demodulation is achieved by multiplying the received signal by a cosine and sine terms to obtain first the baseband equivalent of the received signal, i.e. the in-phase and quadrature signal and thereafter differentially decoding the bits that correspond to the phase shifts of the baseband signal. The studied receiver structure is depicted in Fig.2. In binary DPSK modulation the phase shift of more than ±90 degrees is considered as '1', otherwise '0' is received. However, a frequency error introduces rotation to the baseband signal and thus without compensation the decision margin is reduced to less than ±90 degrees (see Fig. 3). The methods presented here are suitable also for BPSK system, but due to the phase ambiguity introduced by the frequency hopping DPSK modulation was chosen.

In the following, the complex samples of consecutive differentially encoded bits are denoted by $b_k \in [-1,1]$. The received signal over K bit intervals, i.e. over the frequency hop, is considered as

$$r(t) = \sqrt{2S}\sum_{k=1}^{K} b_k p\left(t - nT_c\right)\cos\left([\omega_c + \Delta\omega_d]t + \phi\right) * h(t) + n(t) \tag{1}$$

where $n(t)$ is the background noise of white gaussian distribution, $p(t)$ the bit waveform and $h(t)$ the channel impulse response. In this paper the assumption is made that the channel is one path fixed channel and hence $h(t) = \delta(t)$, i.e. there is no inter symbol

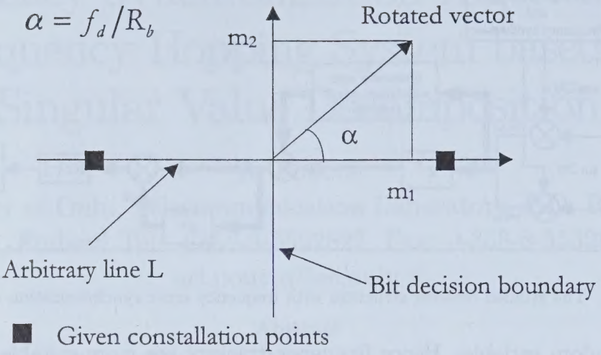


Fig. 3. Signal phasor diagram with normalized frequency error $\alpha = \Delta f_d / R_b$

interference (ISI), and that the phase error $\phi \cong 0$ is accomplished by a reference bit as the first bit in each frequency hop. The frequency error is defined as $\Delta\omega_d\left[l\right] = l2\pi\Delta f_d$, $l = 0, ..., K-1$, where $\Delta f_d = f_c^i - f_0$ is the frequency difference between the local oscillator (f_c^i)and the received signal (f_0). The frequency error presents itself in baseband as rotation of the bit samples. Assuming that the receiver has a bit waveform matched filter $p(-t)$ and one complex sample per bit at correct timing (time synchronicity), the sampled low pass equivalent of the received signal is

$$\overline{\mathrm{r}}_{LP}^{1\times K} = \mathrm{diag}\left(\sqrt{2S}\overline{\mathrm{b}}^{K\times 1} e^{\lceil -j\Delta\overline{\omega}_d^{1\times K}\rceil}\right) + \overline{\mathrm{n}}^{1\times K} \tag{2}$$

which is the input to the frequency error estimator. The superscripts denote the dimensions of a matrix or a vector and $\lceil\cdot\rceil$ denote vector elementwise exponentiation. The traditional way for estimating of the frequency error has included calculating the angular difference of consecutive bit sample vectors (approximate the derivative). These calculated angles are filtered to yield a more accurate estimate of the frequency error. When extracting the frequency error from the *data modulated signal* first the data modulation must be removed. This is accomplished by squaring the bit sample vectors [6] (non-coherent estimation). The vector element-wise squaring gives us vector

$$\overline{\mathrm{r}}_{SQ}^{1\times K} = \mathrm{diag}\left(2S\overline{1}^{K\times 1} e^{\lceil -j2\Delta\overline{\omega}_d^{1\times K}\rceil}\right) + \overline{\mathrm{z}}^{1\times K} \tag{3}$$

The squaring process doubles the angle between the consecutive bit samples, removes the modulation due to $\overline{\mathrm{b}} \odot \overset{\frown}{\overline{\mathrm{b}}} \approx 1$ and squares the amplitude. The operator $\odot$ represents elementwise multiplication of vector elements. The vector $\overline{\mathrm{z}}$ represents the modified noise process introduced by the squaring. In the *pilot channel case* one uses either unmodulated carrier or known data pattern in every M^{th} frequency hop. The unmodulated carrier is equivalent to having $\overline{\mathrm{b}}=\overline{1}$. Assuming time synchronicity, known data pattern allows to write $\overline{\mathrm{b}} \odot \overline{\mathrm{b}} \equiv 1$. Both of the pilot channel approaches allow the squaring to be omitted and hence the frequency error estimates can be extracted from $\overline{\mathrm{r}}_{LP}$.

III. FREQUENCY ERROR ESTIMATION AND CORRECTION ALGORITHMS

The models (2) and (3) correspond to a sinusoidal in noise and hence the problem can be classified as sinusoidal parameter estimation in noise. However, we do not need the parameters as such but a complex conjugate of a replica of the waveworm is sufficient to achieve frequency synchronization. To obtain such a replica of the signal the following is performed. Assuming that the frequency error estimation is performed using the squared signal $\bar{\mathbf{r}}_{SQ}$ (data modulated signal) the frequency error signal replica can be 'estimated' by firstly arranging the samples $\bar{\mathbf{r}}_{SQ}$ into a forward backward linear prediction (FBLP) matrix form [1]

$$\mathbf{A} = \begin{bmatrix} \bar{\mathbf{r}}^*_{SQ}(P-M+1) & \ldots & \bar{\mathbf{r}}^*_{SQ}(1) \\ \bar{\mathbf{r}}^*_{SQ}(P-M+2) & \ldots & \bar{\mathbf{r}}^*_{SQ}(2) \\ \ldots & \ldots & \ldots \\ \bar{\mathbf{r}}^*_{SQ}(P) & \ldots & \bar{\mathbf{r}}^*_{SQ}(M) \\ \bar{\mathbf{r}}_{SQ}(1) & \ldots & \bar{\mathbf{r}}_{SQ}(P-M+1) \\ \bar{\mathbf{r}}_{SQ}(2) & \ldots & \bar{\mathbf{r}}_{SQ}(P-M+2) \\ \ldots & \ldots & \ldots \\ \bar{\mathbf{r}}_{SQ}(M) & \ldots & \bar{\mathbf{r}}_{SQ}(P) \end{bmatrix}, \tag{4}$$

where $\bar{\mathbf{r}}^*_{SQ}$ means complex conjugate of $\bar{\mathbf{r}}_{SQ}$, P denotes total number of samples in K bits assuming in this paper one complex sample per bit ($P = K$). For best performance of the FBLP algorithm, Haykin [1] suggests the value

$$M \approx \frac{P}{3}$$

which is used through out the simulations. In the pilot channel approach one would use $\bar{\mathbf{r}}_{SQ} \rightarrow \bar{\mathbf{r}}_{LP}$. Notice that by reducing the length of the data vector $\bar{\mathbf{r}}_{SQ}$ ($\bar{\mathbf{r}}_{LP}$) by choosing K to be just a part of the whole frequency hop the computation time of SVD can be efficiently decreased. Secondly, the SVD of the matrix $\mathbf{A}$ can be presented by the product of three matrices

$$\mathbf{A}^{m \times n} = \mathbf{U}^{m \times m}\ \Sigma^{m \times n}\ \mathbf{V}^{H\ n \times n} \tag{5}$$

Dimensions of each matrix are written in matrix superscripts. The matrices $\mathbf{U}$ and $\mathbf{V}$ are unitary containing the left and right singular vectors of $\mathbf{A}$ and Σ is a diagonal matrix containing singular values of $\mathbf{A}$. Singular values are conventionally in decreasing order. The operator H is used to denote the complex conjugate transpose (Hermitean) of a matrix. Thirdly, according to Eckhart-Young Theorem [2] the best r_s rank least squares estimate $\widehat{\mathbf{A}}$ (= estimated frequency error signal) for the given data matrix $\mathbf{A}$ is obtained as follows

$$\widehat{\mathbf{A}}^{m \times n} = \sum_{k=1}^{r_s} \sigma_k u_k v_k^* \tag{6}$$

where r_s is the rank of matrix $\widehat{\mathbf{A}}$, σ_k is k'th largest diagonal element (singular value) of matrix Σ, u_k and v_k^* are the k'th columns of the matrixes $\mathbf{U}$ and $\mathbf{V}^*$. The idea is that given signal $\bar{\mathbf{r}}_{SQ}$ (sinusoidal with cyclostationary properties in noise) it suffices to use

$$r_s = 1 \tag{7}$$

to estimate the waveform introduced by the frequency error. The rank reduced matrix $\widehat{\mathbf{A}}$ representing the frequency error and noise is rearranged back to sampled vector form by taking averages of the matrix elements (diagonally) which correspond to the same received sample to obtain an correction signal $\overline{\mathbf{d}}_{2\widehat{\alpha}}$. This signal is related to $\overline{\mathbf{r}}_{SQ}$ and to obtain a proper correction signal $\overline{\mathbf{d}}_{\widehat{\alpha}}$ with respect to $\overline{\mathbf{r}}_{LP}$ the following is needed

$$\overline{\mathbf{d}}_{\widehat{\alpha}} = \exp\left(\frac{j\arg\left(\overline{\mathbf{d}}_{2\widehat{\alpha}}\right)}{2}\right) \tag{8}$$

One should notice that when using pilot channel estimation from $\overline{\mathbf{r}}_{LP}$ one obtains directly $\overline{\mathbf{d}}_{\widehat{\alpha}}$ and Eq. (8) is omitted. Given the signal $\overline{\mathbf{d}}_{\widehat{\alpha}}$ which models the frequency error, the frequency synchronized signal $\overline{\mathbf{c}}$ is obtained as

$$\overline{\mathbf{c}}=\overline{\mathbf{r}}_{\mathbf{LP}} \odot \overline{\mathbf{d}}_{\widehat{\alpha}}^{*} \tag{9}$$

The algorithm described in Eqs. (4)-(9) can be considered as the best frequency error signal correlator in LS-sense, i.e. the singular vectors U_1 and V_1^* span the frequency error signal subspace.

This novel frequency estimator was compared with the averaging filtering approach described e.g. in [3]. The estimate of the frequency error is in the data modulation case

$$2\widehat{\alpha} = \frac{1}{K-1}\sum_{n=2}^{K} \arg[\overline{\mathbf{r}}_{SQ}(n), \overline{\mathbf{r}}_{SQ}(n-1)] \tag{10}$$

Given the frequency error estimate $\widehat{\alpha}$ the corrected bit samples $\overline{\mathbf{c}}$ are obtained as

$$\begin{aligned} \overline{\mathbf{c}} &= \mathrm{Re}\{\overline{\mathbf{r}}_{LP}(k)\}\cos\left[(k-1)\,\widehat{\alpha}\right] + \mathrm{Im}\{\overline{\mathbf{r}}_{LP}(k)\}\sin\left[(k-1)\,\widehat{\alpha}\right] \\ &+j\left(\mathrm{Im}\{\overline{\mathbf{r}}_{LP}(k)\}\cos\left[(k-1)\,\widehat{\alpha}\right] - \mathrm{Re}\{\overline{\mathbf{r}}_{LP}(k)\}\sin\left[(k-1)\,\widehat{\alpha}\right]\right)\ . \end{aligned} \tag{11}$$

in which $k = 1, ..., K$. The third compared method calculates the frequency error estimate as [4]

$$\begin{aligned} 2\widehat{\alpha} = \frac{1}{K-1}\mathrm{sgn}\left\{\begin{matrix}\mathrm{MED} \\ n = 2...K\end{matrix} \arg[\overline{\mathbf{r}}_{SQ}(n), \overline{\mathbf{r}}_{SQ}(n-1)]\right\} \\ \cdot\sum_{n=2}^{K} \left|\arg[\overline{\mathbf{r}}_{SQ}(n), \overline{\mathbf{r}}_{SQ}(n-1)]\right| \end{aligned} \tag{12}$$

and uses Eq. (11) for frequency synchronization.

In [4] the pdf for $\alpha = \frac{\Delta f_d}{R_b}$ was given as

$$\begin{aligned} p\left(\alpha\right) = \frac{e^{-\frac{m_1^2+m_2^2}{2\sigma^2}}}{2\pi} \cdot \\ \left[e^{2\lambda^2} + \sqrt{4\pi}\lambda e^{\lambda^2}\left(1 - \frac{1}{2}\,\mathrm{erf\,c}\left[\lambda\right]\right)\right] \end{aligned} \tag{13}$$

with $\lambda = \frac{(m_1\cos\alpha + m_2\sin\alpha)}{\sqrt{2}\sigma}$. The means m_1 and m_2 are assumed to be stationary within a frequency hop and hence the means are

$$\begin{aligned} m_1 &= \sqrt{\frac{E_b}{N_0}}\cos\alpha \\ m_2 &= \sqrt{\frac{E_b}{N_0}}\sin\alpha \end{aligned} \tag{14}$$

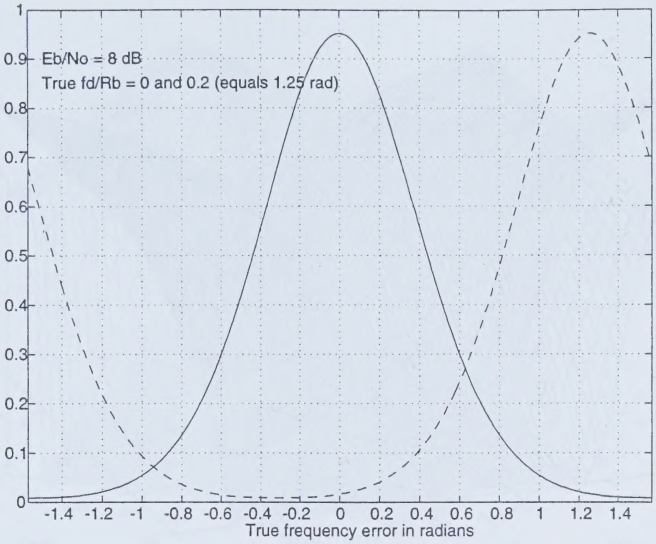


Fig. 4. The probability density function for $\Delta f_d/R_b = 0$ and $\Delta f_d/R_b = 0.2$ with $E_b/N_0 = 8$ dB for data modulated signal ($\sigma^2 = 2E_bN_0 + N_0^2$ resulting from the squaring)

in which R_b is the bit rate, E_b the bit energy and N_0 is one-sided power spectral density of the gaussian noise. Estimator in Eq. (10) is optimal in the sense that it achieves the Cramer-Rao bound for $\alpha \sim 0$ with introduction of weighting coefficients [3]. The pdf in Eq. (13) is close to Gaussian when $\alpha = \frac{\Delta f_d}{R_b} \sim 0$ as can be seen from Fig. 4. When $|\alpha| = \left|\frac{\Delta f_d}{R_b}\right| \rightarrow \frac{1}{4}$, however, the estimator given in Eq. (12) was found to outperform (10) [4] since the pdf of α is deviated from the Gaussian pdf which is due to a phenomenon known as the equivocation [5] or cycle slipping [6]. The analysis of SVD-estimator, as well as the estimator in Eq. (12), is difficult due to the nonlinear nature of the estimators. The relative performances of the estimators were hence obtained by simulations.

IV. NUMERICAL RESULTS

The simulations were performed only for the data modulated case which in general is the more difficult scenario. In Figs. 5, 6 and 7 the performance of the averaging, the SVD based and the absolute averaging estimators are presented. The performance measure is the mean squared error $MSE = E\left\{\left(\frac{\Delta\hat{f}_d - \Delta f_d}{R_b}\right)^2\right\}$ of the frequency error estimate as a fuction of E_b/N_0 and $\Delta f_d/R_b$. Each simulation point represent the MSE calculated over 1 000 frequency hops with 10 bits transmitted on each hop. One can draw several conclusions from the figures. First of all, one can draw the conclusion that the averaging estimator performs very well when the frequency error is small. This was expected result based on the discussion in chapter III. As the E_b/N_0 value increases the operation area in terms of $\Delta f_d/R_b$ increases but remains smaller than that what was achieved for the SVD based and the absolute averaging estimators. The absolute averaging estimator suffers from poor performance when $\Delta f_d/R_b$ is small due to the nonlinear operation ($|\cdot|$) which enhances the noise which again was expected. When $\Delta f_d/R_b$ increases the cycle

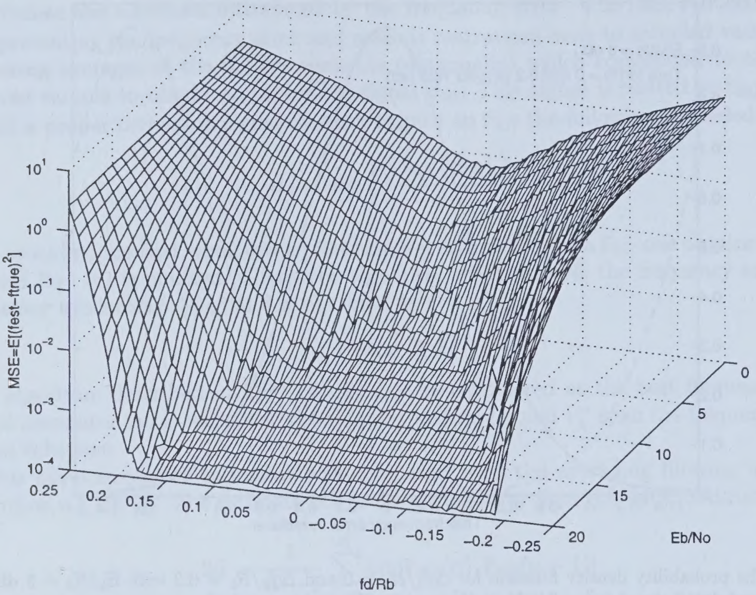


Fig. 5. MSE performance of the averaging estimator as the function of E_b/N_0 and $\Delta f_d/R_b$.

slipping phenomenon becomes dominant factor in the frequency error and due to the median processing the estimator is rather robust in this scenario and hence can tolerate much higher frequency error values than averaging estimator. The SVD based estimator removes the sensitivity to noise when $\Delta f_d/R_b$ is small. The estimator requires, however, a little bit higher E_b/N_0 conditions in order to operate optimally. The overall operation range in $\Delta f_d/R_b$ and E_b/N_0 domain is the largest for the SVD based estimator. The problem associated with SVD approaches is the computational complexity.

One alternative to alleviate the computational burden is to combine the estimators given in Eqs. (10) and (12). The approach is to use (12) for high frequency error and (10) for small frequency error. The problem is then to find a suitable criterion for switching between the two algorithms. By conducting some trial simulations the following simple decision rule was obtained

$$2\widehat{\alpha} = \begin{cases} \text{use (10), when } |\widehat{\alpha}| \text{ given by (12) } \leq \theta_{\text{lim}} \\ \text{use (12), when } |\widehat{\alpha}| \text{ given by (12) } > \theta_{\text{lim}} \end{cases} \tag{15}$$

The value θ_{lim} was optimised for $E_b/N_0 = 8$ dB by simulations. $E_b/N_0 = 8$ dB was chosen as the optimisation point because it is approximately the threshold value where the estimators (10) and (12) still can operate. The obtained value was $\theta_{\text{lim}} = \frac{\pi}{5}$. Figure 8 presents the MSE performance obtained for the combined frequency error estimator. The results show that the operation range is now larger than that obtained for the averaging estimator and the noise enhancement problem near $\Delta f_d/R_b \sim 0$ characteristic to the absolute averaging estimator is also greatly alleviated. The performance obtained with this hybrid estimator is rather close to that obtained by the SVD based estimator.

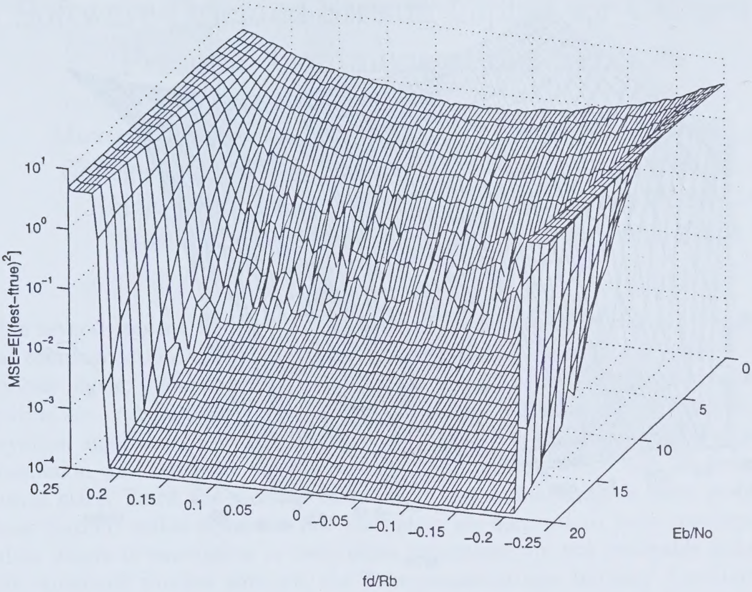


Fig. 6. MSE performance of the SVD based estimator as the function of E_b/N_0 and $\Delta f_d/R_b$.

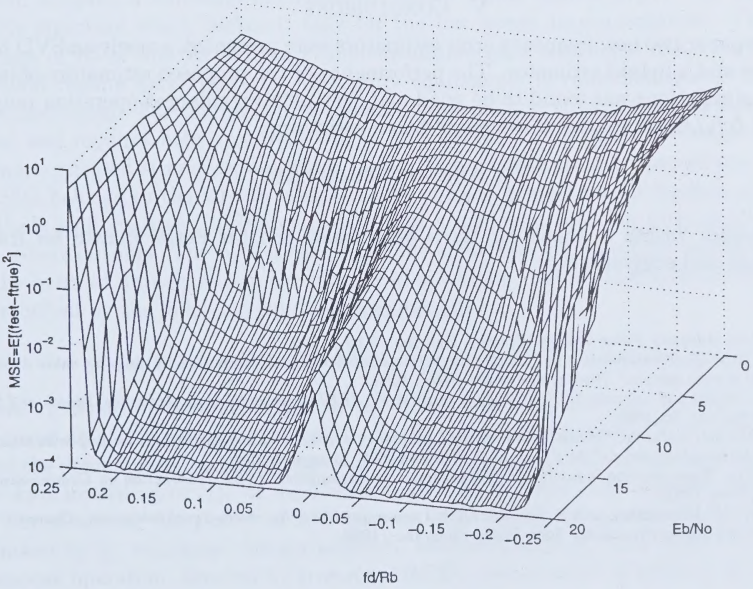


Fig. 7. MSE performance of the absolute averaging estimator as the function of E_b/N_0 and $\Delta f_d/R_b$.

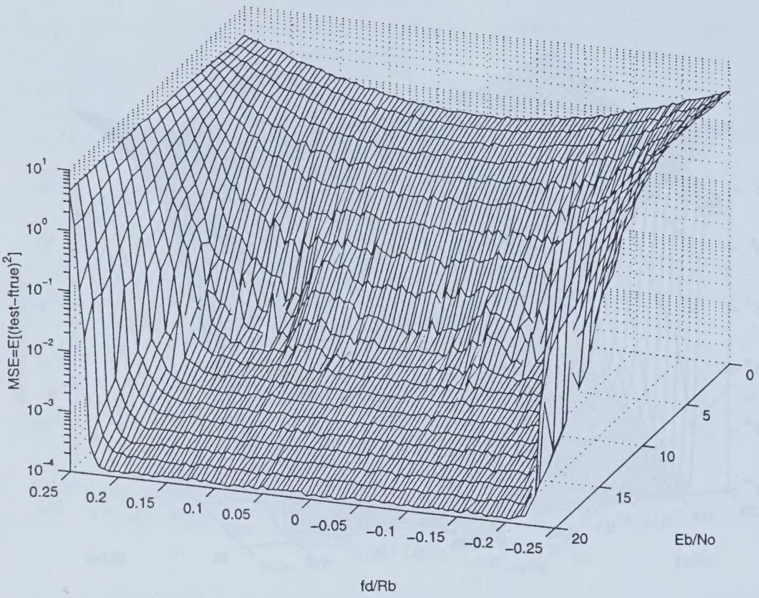


Fig. 8. MSE performance of the combined averaging and absolute averaging estimator as the function of E_b/N_0 and $\Delta f_d/R_b$.

V. Conclusions

In this paper two new frequency error estimators were presented, namely an SVD based estimator and a hybrid estimator. The performance of the proposed estimators obtained through simulations was found to be good and they offer an enlarged operation range in terms of $\Delta f_d/R_b$ compared to averaging based estimator.

VI. Acknowledgments

The author wishes to thank Mr Harri Saarnisaari and Mr Pertti Henttu for fruitful discussions and suggestions.

References

[1] S. Haykin, *Adaptive Filter Theory.* Prentice Hall Inc., 3 ed., 1996.

[2] D. W. Tufts, R. Kumaresan, and I. Kirsteins, "Data adaptive signal estimation by singular value decomposition of a data matrix," *Proceedings of the IEEE*, 1982.

[3] S. Kay, "A fast and accurate single frequency estimator," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 39, 1989.

[4] A. Pouttu and J. Juntti, "Median filter assisted frequency synchronization algorithms for hop-wise estimated frequency hopping system," in *Proceedings of PIMRC'99, to appear*, 1999.

[5] M. P. Fitz, "Equivocation in nonlinear digital carrier synchronizers," *IEEE Transaction on Communications*, vol. 39, Nov. 1991.

[6] H. Meyr, M. Moeneclaey, and S. Fechtel, *Digital communication Receivers-Synchronization, Channnel Estimation and Signal Processing.* John Wiley Sons Inc., 1998.

A Software-Oriented Stream Cipher for Cellular and Personal Communications Services

Muxiang Zhang Agnes Chan
College of Computer Science
Northeastern University
Boston, MA 02115

Christopher Carroll
GTE Laboratories Inc.
40 Sylvan Road
Waltham, MA 02254

For several reasons, encryption algorithms have been constrained in cellular and personal communications. First, the lack of computing power in mobile station, especially in mobile phone, limits the use of computationally intensive encryption algorithms such as public key cryptography; Second, due to the high bit error rate of wireless channels, encryption algorithms which produce error propagation deteriorate the quality of data transmission, and hence are not well suited to applications where high bit error rates are common place; Third, the shortage of bandwidth at uplink channels (from mobile station to base station) makes ciphers at low encryption (or decryption) rates unacceptable, and random delays in encryption or decryption algorithms are not preferable either. In the North American wireless network, the Telecommunications Industry Association (TIA) has proposed several encryption algorithms in the last few years. All these encryption algorithms have not gone through public scrutinizing process and their weaknesses were discovered within the last two years [1,2,3]. The European Group Special Mobile(GSM) system adopted a hardware-implemented stream cipher known as A5. A5 has a very simple structure which makes it easy for the low power implementation. The disadvantage of A5 is that its hardware implementation adds additional cost. Furthermore, it would require exorbitant cost to modify equipments if vulnerabilities of the cipher were discovered. For this reason, a software implemented stream cipher which has fast speed and requires little resources will be preferable. To this end, we designed SSC, a software-oriented stream cipher which is easy to implement in 8-bit processors.

SSC belongs to the stream cipher family of filter generators. It consists of a linear feedback shift register (LFSR) of 16-byte and a nonlinear function with 16-byte inputs and 2-byte outputs. The byte sequences of the keystream generator are added modulo-2 to the bytes of data frames in the manner of a Vernam cipher. The byte-oriented LFSR is described by the following equation:

$$\mathbf{v}_{n+16} = \mathbf{v}_{n+14} \oplus \mathbf{v}_{n+2}\mathbf{A}_{13} \oplus \mathbf{v}_{n+1}\mathbf{A}_{14} \oplus \mathbf{v}_n\mathbf{A}_{15} \quad \text{for } n = 0, 1, \ldots \tag{1}$$

where $\mathbf{v}_n$ denotes a byte generated by the LFSR, $\mathbf{A}_{13}, \mathbf{A}_{14}$, and $\mathbf{A}_{15}$ are 8×8 matrices. To speed up the matrix multiplications in (1), we use three tables $\mathbf{T}_{13}, \mathbf{T}_{14}$ and $\mathbf{T}_{15}$ to store the precomputed values of the matrix multiplications in (1). Thus, to compute each new byte according to (1), we need three table look-ups and three modulo-2 additions.

The nonlinear function f in SSC consists of five kinds of operations, concatenation (denoted by $||$), transpose, integer addition, modulo-2 addition, and table look-up. The transpose operation, denoted by $trans(c, ABCD)$, permutes the 4 bytes A, B, C, and D

in a word . The transpose is controlled by the control bit c. When c is 0, $trans(c, ABCD)$ equals $ABCD$, otherwise $CDAB$. The two tables T_0 and T_1 are constructed from the Galois filed $GF(2^8)$. Let $\mathbf{V}_n = (\mathbf{v}_{n+15}, \mathbf{v}_{n+14}, \ldots, \mathbf{v}_n)$ denote the shift register state, the nonlinear function f is defined by the following pseudo-code:

```
function $f(\mathbf{V}_n)$
$A \leftarrow \mathbf{v}_{n+15}||\mathbf{v}_{n+14}||\mathbf{v}_{n+13}||\mathbf{v}_{n+12},\ c[0] \leftarrow 1$;
for $l \leftarrow 0$ to 2 do
    $B \leftarrow trans(c[l], \mathbf{v}_{n+11-4l}||\mathbf{v}_{n+10-4l}||\mathbf{v}_{n+9-4l}||\mathbf{v}_{n+8-4l})$;
    $A \leftarrow A + B$;
    $c[l+1] \leftarrow$ carry;
$i \leftarrow c[3] \oplus c[1], j \leftarrow c[3] \oplus c[2]$;
return $\mathbf{T}_i(A[0] \oplus A[2])||\mathbf{T}_j(A[1] \oplus A[3])$;
```

SSC supports private keys of various sizes, from 4 bytes to 16 bytes. For a private key of 16 bytes, we can use the private key as the initial state of the LFSR. When the private key is less than 16 bytes, we stretch the private key to a 16-byte for the initial state of the LFSR. This procedure is described as follows:

```
procedure $K_{master}(K, m)$
If $m = 16$ then return $K$;
$\mathbf{V}_0 \leftarrow (K||K||K||K||K)_{128}, t \leftarrow 0$;
//where $(S)_i$ denotes the left most $i$ bits of the string $S$
for $i \leftarrow 0$ to 1 do;
    $A \leftarrow f(\mathbf{V}_t)$;
    for $j \leftarrow 1$ to 7 do
        run the keystream generator one time, $t \leftarrow t + 1$;
        $A \leftarrow A||f(\mathbf{V}_t)$;
    $\mathbf{V}_t \leftarrow A$;
return $K||(\mathbf{V}_t)_{128-m}$;
```

The LFSR, as well as the nonlinear filtering function, was designed such that the cipher is not vulnerable to known correlation attacks and that the generated sequence has good statistical properties, such as long period and large linear complexity.

References

[1] David Wagner, Bruce Schneier and John Kelsey, "Cryptanalysis of the cellular message encryption algorithms", *Proceedings of Crypto'97, LNCS 1294*, Burton S. Kaliski Jr. Ed., Springer-Verlag, 1997.

[2] D. Wagner, L. Simpson, E. Dawson, John Kelsey, W. Millan, and B. Schneier, "Cryptanalysis of ORYX", *Proceedings of Selected Areas in Cryptography*, 1998.

[3] M. Zhang, C. Carroll, and A. Chan, "Analysis of IS-95 CDMA Voice Privacy", *Technical Memorandum, GTE Laboratories Inc.*, TM-1011-09-97-589, 1997.

Binary Rank Criteria for PSK Modulated Space-Time Codes*

Hesham El Gamal and A. Roger Hammons, Jr

Abstract

One of the fundamental difficulties of space-time codes, which has hindered the development of more general results, is the fact that the diversity advantage design criterion applies to the complex domain of the base-band modulated signals rather than the binary or discrete domain in which the underlying codes are traditionally designed. In this paper, we develop general binary rank criteria, pertaining to the unmodulated code words (or their projections)–as binary matrices over the binary field–that guarantee that the space-time code achieves full spatial diversity for BPSK and QPSK modulations. These results unify much of what is currently known about space-time code design and lead to new, more sophisticated construction methods.

1 Introduction

Reliable communication over the fading wireless channel is made possible only through the use of diversity techniques in which the receiver is afforded multiple opportunities to observe the signal under varying channel conditions. Commonly used methods include: 1) Frequency diversity, in which the signal is transmitted on multiple RF carriers; 2) Temporal diversity, in which channel coding and interleaving are used to replicate and distribute the signal in time; and 3) Spatial diversity, in which multiple spatially separated antennas are used either at the transmitter or the receiver to provide multiple replicas of the signal with decorrelated fading characteristics.

In a simple transmit diversity scheme, a delayed replica of the transmitted signal might be retransmitted through a second, spatially-independent antenna with the two signals being coherently combined at the receiver. From a coding perspective, this transmit diversity scheme amounts to a repetition code and leads to the natural question as to whether more sophisticated codes might be more effective. Information-theoretic studies by Teletar [1] and Foschini and Gans [2] have shown that the capacity of multi-antenna systems significantly exceeds that of conventional single-antenna systems in fading channels. The challenge of designing channel codes for high capacity multi-antenna systems has led to the development of the so called "space-time codes" in which coding is performed across the spatial dimension as well as time [3] [4].

2 Background

For the space-time communication system under consideration, we have the following model. At the transmitter, the source generates k information symbols from the discrete alphabet $\mathcal{X}$, which are encoded by the error control code C to produce code words of length $N = nL_t$ over the symbol alphabet $\mathcal{Y}$. The encoded symbols are parsed among L_t transmit antennas and then mapped by the modulator into constellation points from the discrete complex-valued signaling constellation Ω for transmission across the channel. The modulated streams for all antennas are transmitted simultaneously.

At the receiver, there are L_r receive antennas to collect the incoming transmissions. The received baseband signals are subsequently decoded by the space-time decoder. Each spatial channel (the link between one transmit antenna and one receive antenna) is assumed to experience statistically independent flat Rayleigh fading. Receiver noise is assumed to be additive white Gaussian noise (AWGN).

Since we are interested in algebraic code designs, we formally define a space-time code to consist of an underlying error control code together with the spatial parsing format.

Definition 1 *An $L \times n$ space-time code $\mathcal{C}$ of size M consists of an (Ln,M) error control code C and a spatial parser σ that maps each code word vector $\bar{c} \in C$ to an $L \times n$ matrix $\mathbf{c}$ whose entries are a rearrangement of those of $\bar{c}$. The space-time code $\mathcal{C}$ is said to be linear if both C and σ are linear.*

In this paper, it suffices to consider the standard parser that maps

$$\bar{c} = (c_1^1, c_1^2, \ldots, c_1^{L_t}, \ldots, c_n^1, c_n^2, \ldots, c_n^{L_t}) \in C$$

to the matrix

$$\mathbf{c} = \begin{bmatrix} c_1^1 & c_2^1 & \cdots & c_n^1 \\ c_1^2 & c_2^2 & \cdots & c_n^2 \\ \vdots & \vdots & \ddots & \vdots \\ c_1^{L_t} & c_2^{L_t} & \cdots & c_n^{L_t} \end{bmatrix}.$$

In this notation, it is understood that c_t^i is the code symbol assigned to transmit antenna i at time t. As discussed in [9], other parsings may also be of interest.

Let $f : \mathcal{Y} \rightarrow \Omega$ be the modulator mapping function. Then $\mathbf{s} = f(\mathbf{c})$ is the baseband version of the code word as transmitted across the channel. For this system, we have the following baseband model of the received signal:

$$y_t^j = \sum_{i=1}^{L_t} \alpha_{ij} s_t^i \sqrt{E_s} + n_t^j, \qquad (1)$$

where y_t^j is the signal received at antenna j at time t; α_{ij} is the complex path gain from transmit antenna i to receive antenna j; $s_t^i = f(c_t^i)$ is the transmitted constellation point corresponding to c_t^i; and n_t^j is the AWGN noise sample for receive antenna j at time t. The noise samples are independent samples of a zero-mean complex Gaussian random variable with variance $N_0/2$ per dimension. The fading channel is quasi-static in the sense that, during the transmission of n code word symbols across any one of the links, the complex path gains do not change with time t, but are independent from one code word transmission to the next.

From [4] [3], the fundamental performance parameters for space-time codes are well known to be: (1) Diversity advantage, which describes the exponential decrease of decoded error rate versus signal-to-noise ratio (asymptotic slope of the performance curve in a log-log scale); and (2) Coding advantage which does not affect the asymptotic slope but results in a shift in the performance curve. This leads to the design criteria:

- *Rank Criterion*: Maximize the diversity advantage $r = \text{rank}(f(\mathbf{c}) - f(\mathbf{e}))$ over all pairs of distinct code words $\mathbf{c}, \mathbf{e} \in \mathcal{C}$.

- *Product Distance Criterion*: Maximize the coding advantage $\eta = (\lambda_1 \lambda_2 \cdots \lambda_r)^{1/r}$, which is the geometric mean of the nonzero eigenvalues of $\mathbf{A} = (f(\mathbf{c}) - f(\mathbf{e}))(f(\mathbf{c}) - f(\mathbf{e}))^{\mathrm{H}}$, over all pairs of distinct code words $\mathbf{c}, \mathbf{e} \in \mathcal{C}$.

The rank criterion is the more important of the two as it determines the asymptotic slope of the performance curve as a function of E_s/N_0. For an $L \times n$ space-time code $\mathcal{C}$, the maximum possible rank is L. Consequently, full spatial diversity is achieved if all baseband difference matrices corresponding to distinct code words in $\mathcal{C}$ have full rank L.

In [4], Tarokh *et al.* proposed simple design rules for space-time trellis codes for $L = 2$ spatial diversity:

> *Rule 1.* Transitions departing from the same state differ only in the second symbol.
>
> *Rule 2.* Transitions merging at the same state differ only in the first symbol.

Two good codes that satisfy these design rules, and a few others that do not, were handcrafted using computer search methods.

In [5], Grimm *et al.* introduced the notion of "zeroes symmetry" as a generalization of the Tarokh design rules for higher levels of diversity $L \geq 2$. A space-time code has zeroes symmetry if every baseband code word difference $f(\mathbf{c}) - f(\mathbf{e})$ is upper and lower triangular (and has appropriate nonzero entries to ensure full rank). As noted in [5], the zeroes symmetry property is sufficient for full rank but not necessary; nonetheless, it is useful in constraining computer searches for good space-time codes. Grimm *et al.* presented the results of a significant computer search undertaken to identify full diversity space-time codes with good coding advantage. They provide a small table of short constraint length space-time trellis codes that achieve full spatial diversity ($L = 2$, 3, and 5 for BPSK modulation; $L = 2$ for QPSK modulation). They discuss the difficulties involved in evaluating coding advantage for general space-time trellis codes.

The general design of non-trivial space-time codes is hampered by the fact that the rank criterion applies to the complex-valued differences between the baseband versions of the code words. It is not easy to transfer this design criterion into the binary domain where the problem of code design is relatively well understood. In the next section, we develop general binary design criteria that are sufficient to guarantee that a space-time code achieves full spatial diversity.

3 Binary Rank Criteria

In the rank criterion for space-time codes, the sign of the differences between modulated code word symbols is important. On the other hand, it is difficult to see how one would be able to preserve that information in the binary domain. Thus, we are led to investigate what can be said in the absence of such knowledge.

Definition 2 *Two complex matrices* $\mathbf{r}_1$ *and* $\mathbf{r}_2$ *will be said to be* ω*-equivalent if* $\mathbf{r}_1$ *can be transformed into* $\mathbf{r}_2$ *by multiplying any number of entries of* $\mathbf{r}_1$ *by powers of the complex number* ω.

For BPSK, we are interested in (-1)-equivalence. For QPSK, we are interested in i-equivalence where $i = \sqrt{-1}$. The following special notation will be used:

BPSK : $\doteq$ denotes the (-1)-equivalence.
QPSK : $\mathrel{\overset{..}{=}}$ denotes the i-equivalence.

Using this notion, we derive binary rank criteria for space-time codes that depend only on the unmodulated code words themselves. The binary rank criterion provides a complete characterization for BPSK-modulated codes (under the assumption of lack of knowledge regarding signs in the baseband differences). It provides a highly effective characterization for QPSK-modulated codes.

I. BPSK-Modulated Codes

For BPSK modulation, the natural discrete alphabet is the field $\mathbb{F} = \{0, 1\}$ of integers modulo 2. Modulation is performed by mapping the symbol $x \in \mathbb{F}$ to the constellation point $s = f(x) \in \{-1, 1\}$ according to the rule $s = (-1)^x$. Note that it is possible for the modulation format to include an arbitrary phase offset $e^{i\phi}$, since a uniform rotation of the constellation will not affect the rank of the matrices $f(\mathbf{c}) - f(\mathbf{e})$ nor the eigenvalues of the matrices

$A = (f(c) - f(e))(f(c) - f(e))^H$. Notationally, the circled operator $\oplus$ will be used to distinguish modulo 2 addition from real- or complex-valued $(+, -)$ operations.

Theorem 3 *Let C be a linear $L \times n$ space-time code with $n \geq L$. Suppose that every non-zero binary code word $\mathbf{c} \in C$ has the property that every real matrix (-1)-equivalent to $\mathbf{c}$ is of full rank L. Then, for BPSK transmission, C satisfies the space-time rank criterion and achieves full spatial diversity L.*

Proof: It is enough to note that $[(-1)^{c_1} - (-1)^{c_2}]/2 \doteq c_1 \oplus c_2$. □

It turns out that (-1)-equivalence has a simple binary interpretation.

Theorem 4 *The $L \times n$, $(n \geq L)$, binary matrix $\mathbf{c} = [\ \bar{a}_1\ \bar{a}_2\ \cdots\ \bar{a}_L\ ]^T$ has full rank L over the binary field $\mathbb{F}$ if and only if every real matrix $\mathbf{r} = [\ \bar{r}_1\ \bar{r}_2\ \cdots\ \bar{r}_L\ ]^T$ that is (-1)-equivalent to $\mathbf{c}$ has full rank L over the real field $\mathbb{R}$.*

Proof: Follows directly from consideration of the real and binary versions of $\alpha_1 \bar{r}_1 + \alpha_2 \bar{r}_2 + \cdots + \alpha_L \bar{r}_L = 0$. See [9] [10] for details. □

The binary design criterion for linear space-time codes now follows immediately.

Theorem 5 (Binary Rank Criterion) *Let C be a linear $L \times n$ space-time code with $n \geq L$. Suppose that every non-zero binary code word $\mathbf{c} \in C$ is a matrix of full rank over the binary field $\mathbb{F}$. Then, for BPSK transmission, the space-time code C achieves full spatial diversity L.*

Although the binary rank criterion and associated theorems are stated for linear codes, they also work for nonlinear codes, provided the results are applied to the modulo 2 differences between code words instead of the code words themselves.

II. QPSK-Modulated Codes

For QPSK modulation, the natural discrete alphabet is the ring $\mathbb{Z}_4 = \{0, \pm 1, 2\}$ of integers modulo 4. Modulation is performed by mapping the symbol $x \in \mathbb{Z}_4$ to the constellation point $s \in \{\pm 1, \pm i\}$ according to the rule $s = i^x$, where $i = \sqrt{-1}$. Again, the absolute phase reference of the QPSK constellation could have been chosen arbitrarily without affecting the diversity advantage or coding advantage of a $\mathbb{Z}_4$-valued space-time code.

For the $\mathbb{Z}_4$-valued matrix $\mathbf{c}$, we define the binary component matrices $\alpha(\mathbf{c})$ and $\beta(\mathbf{c})$ to satisfy the expansion

$$\mathbf{c} = \beta(\mathbf{c}) \oplus_4 2\alpha(\mathbf{c}).$$

Thus, $\beta(\mathbf{c})$ is the modulo 2 projection of $\mathbf{c}$ and $\alpha(\mathbf{c}) = [\mathbf{c} \ominus_4 \beta(\mathbf{c})]/2$. Here $\oplus_4$ and $\ominus_4$ denote addition and subtraction modulo 4.

We also introduce the following special matrices that are useful in the analysis of QPSK-modulated space-time codes:

- Complex-valued $\zeta(\mathbf{c}) = \mathbf{c} + i\beta(\mathbf{c})$.
- Binary-valued indicant projections: $\Xi(\mathbf{c})$ and $\Psi(\mathbf{c})$.

The indicant projections to be defined below serve to indicate certain aspects of the binary structure of the $\mathbb{Z}_4$ matrix in which multiples of two are ignored.

A $\mathbb{Z}_4$-valued matrix $\mathbf{c}$ of dimension $L \times n$ will be said to be of *type* $1^\ell 2^{L-\ell} \times 1^m 2^{n-m}$ if it consists of exactly ℓ rows and m columns that are not multiples of two. When the column (row) structure of a matrix is not of particular interest, the matrix will be said to be of *row type*$1^\ell 2^{L-\ell}$ (*column type*$1^m 2^{n-m}$).

Let $\mathbf{c}$ be a $\mathbb{Z}_4$-valued matrix of type $1^\ell 2^{L-\ell} \times 1^m 2^{n-m}$. After suitable row permutations if necessary, it has the following row structure:

$$\mathbf{c} = \begin{bmatrix} \bar{c}_1 \\ \vdots \\ \bar{c}_\ell \\ 2\bar{c}'_{\ell+1} \\ \vdots \\ 2\bar{c}'_L \end{bmatrix}.$$

The row-based indicant projection (Ξ-projection) is then defined as

$$\Xi(\mathbf{c}) = \begin{bmatrix} \beta(\bar{c}_1) \\ \vdots \\ \beta(\bar{c}_\ell) \\ \beta(\bar{c}'_{\ell+1}) \\ \vdots \\ \beta(\bar{c}'_L) \end{bmatrix}.$$

Similarly, the column-based indicant projection (Ψ-projection) is defined as

$$[\Psi(\mathbf{c})]^{\mathrm{T}} = \Xi(\mathbf{c}^{\mathrm{T}}). \tag{2}$$

The first result shows that the baseband difference of two QPSK-modulated code words is directly related to the $\mathbb{Z}_4$-difference of the unmodulated code words.

Proposition 6 *Let C be a $\mathbb{Z}_4$ space-time code. For $\mathbf{x}, \mathbf{y} \in C$, let $i^{\mathbf{x}} - i^{\mathbf{y}}$ denote the baseband difference of the corresponding QPSK-modulated signals. Then,*

$$i^{\mathbf{x}} - i^{\mathbf{y}} \doteq \zeta(\mathbf{x} \ominus_4 \mathbf{y}).$$

Proof: Each entry $i^x - i^y$ can be turned into the complex number $(x \ominus_4 y) + i(x \oplus y)$ by multiplying by ± 1 or $\pm i$ as necessary. □

Theorem 7 *Let C be a linear, $L \times n$ $(n \geq L)$ space-time code over $\mathbb{Z}_4$. Suppose that every non-zero code word $\mathbf{c} \in C$ has the property that every complex matrix i-equivalent to $\zeta(\mathbf{c})$ is of full rank L. Then, for QPSK transmission, C satisfies the space-time rank criterion and achieves full spatial diversity L.*

It turns out that the indicant projections of the $\mathbb{Z}_4$-valued matrix $\mathbf{c}$ provide a great deal of information regarding the singularity of $\zeta(\mathbf{c})$ and any of its i-equivalents. Thus, the indicants provide the basis for our binary rank criterion for QPSK-modulated space-time codes.

Theorem 8 *Let $\mathbf{c} = [\ \bar{c}_1\ \ \bar{c}_2\ \ \cdots\ \ \bar{c}_L\ ]^{\mathrm{T}}$ be an $\mathbb{Z}_4$-valued matrix of dimension $L \times n$, $(n \geq L)$. If the row-based indicant $\Xi(\mathbf{c})$ or the column-based indicant $\Psi(\mathbf{c})$ has full rank L over $\mathbb{F}$, then every complex matrix $\mathbf{z}$ that is i-equivalent to $\zeta(\mathbf{c})$ has full rank L over the complex field $\mathbb{C}$.*

Proof: The proof for this Theorem is rather involved and the reader is referred to [9] or [10] for a detailed proof. □

The binary rank criterion for QPSK space-time codes now follows as an immediate consequence of the previous two theorems.

Theorem 9 (QPSK Binary Rank Criterion I) *Let C be a linear $L \times n$ space-time code over $\mathbb{Z}_4$, with $n \geq L$. Suppose that, for every non-zero $\mathbf{c} \in C$, the row-based indicant $\Xi(\mathbf{c})$ or the column-based indicant $\Psi(\mathbf{c})$ has full rank L over $\mathbb{F}$. Then, for QPSK transmission, the space-time code C achieves full spatial diversity L.*

In certain $\mathbb{Z}_4$ space-time code constructions, there may be no code word matrices having isolated rows or columns that are multiples of two. (It is always possible of course for the entire code word to be a multiple of two.) In this case, the following binary rank criterion is simpler yet sufficient.

Theorem 10 (QPSK Binary Rank Criterion II) *Let C be a linear $L \times n$ space-time code over $\mathbb{Z}_4$, with $n \geq L$. Suppose that, for every non-zero $\mathbf{c} \in C$, the binary matrix $\beta(\mathbf{c})$ is of full rank over $\mathbb{F}$ whenever $\beta(\mathbf{c}) \neq 0$, and $\beta(\mathbf{c}/2)$ is of full rank over $\mathbb{F}$ otherwise. Then, for QPSK transmission, the space-time code C achieves full spatial diversity L.*

Proof: Under the specified assumptions, we have either $\Xi(\mathbf{c}) = \beta(\mathbf{c})$ or $\Xi(\mathbf{c}) = \beta(\mathbf{c}/2)$, depending on whether $\beta(\mathbf{c}) = 0$ or not. □

4 Algebraic Code Design

The binary rank criteria lead to general algebraic code constructions for full diversity space-time trellis and block codes suitable for arbitrary number of transmit antennas. The reader is referred to [9] [10] for detailed development.

5 Conclusions

The binary rank criteria greatly simplify the design of space-time codes because the code design problem is returned back to the binary domain where much is understood about designing good codes. In the absence of such design criteria, one is restricted to computer-assisted handcrafting of codes according to ad hoc design rules or to exhaustive computer searches for codes of a particular limited structure. The binary rank criteria make it is possible to develop algebraic code designs for which one can prove that full spatial diversity is achieved without resorting to brute-force verification. It is worth noting that the binary rank criteria for BPSK and QPSK space-time codes apply to both trellis and block codes and for arbitrary numbers of transmit antennas. In [9] we show that all but one of the BPSK and QPSK-modulated space-time codes found thus far fall within the scope of our theory. The binary rank criteria open the door to more sophisticated code designs.

References

[1] E. Teletar. Capacity of multi-antenna gaussian channels. *Technical Report, AT&T-Bell Labs*, June 1995.

[2] G. J. Foschini and M. Gans. On the limits of wireless communication in a fading environment when using multiple antennas. *Wireless Personal Communication*, 6:311–335, Mar 1998.

[3] J.-C. Guey, M. R. Bell M. P. Fitz, and W.-Y. Kuo. Signal design for transmitter diversity wireless communication systems over Rayleigh fading channels. *IEEE Vehicular Technology Conference*, pages 136–140, Atlanta, 1996.

[4] V. Tarokh, N. Seshadri, and A. R. Calderbank. Space-time codes for high data rate wireless communication: Performance criterion and code construction. *IEEE Trans. Info. Theory*, IT-44:774–765, March 1998.

[5] J. Grimm, M. P. Fitz, and J. V. Krogmeier. Further results in space-time coding for Rayleigh fading. *36th Allerton Conference on Communications, Control and Computing proceedings*, September 1998.

[6] S. M. Alamouti. A simple transmitter diversity scheme for wireless communications. *IEEE Journal on Selected Areas in Communications*, October 1998.

[7] V. Tarokh, H. Jafarkhani, and A. R. Calderbank. Space-time block codes from orthogonal designs. *IEEE Trans. Info. Theory*, IT-45:1456–1467, July 1999.

[8] V. Tarokh, H. Jafarkhani, and A. R. Calderbank. Space-time block coding for wireless communications: Performance results. *Preprint*, 1998.

[9] A. R. Hammons Jr. and H. El Gamal. On the theory of space-time codes for PSK modulation. *to appear in IEEE Trans. Info. Theory.*

[10] H. El Gamal. On the theory and applications of space-time and graph based codes. *Ph.D Dissertation, University of Maryland at College Park*, May 1999.

Turbo codes with Orthogonal Modulation in DS-CDMA Mobile Radio System with Short Frame Transmission

Ge Li and Yong Liang Guan
Nanyang Technological University
Nanyang Avenue, Singapore 639798

Abstract— **This paper studies the application of Turbo codes for short frame transmission in direct sequence code division multiple access (DS-CDMA) mobile radio system with M-ary orthogonal signaling and noncoherent reception. When Turbo codes is combined with orthogonal modulation, we actually have a hybrid concatenated coding scheme with the parallel-concatenated Turbo coding as outer coding and orthogonal modulation as inner coding. The proposed method is useful for some applications, such as the uplink of a DS-CDMA system, and has a notable performance advantage over other schemes proposed in the literature. Computer simulations are carried out for short frame transmission (192bits/frame) on a variety of channels, e.g., coherent/noncoherent AWGN channel, multipath Rayleigh fading channel of different fading rates, and closed-loop power controlled channels of different feedback delays. The simulated performances are compared with separate decoding of Turbo codes with orthogonal modulation, and iterative decoding and demodulation of convolutional code with orthogonal modulation. The results show that considerable performance enhancement can be achieved with moderate computational complexity. Also, the best performance is achieved with a randomized, instead of power controlled, fading channel.**

Index terms—**Turbo codes, orthogonal modulation, iterative decoding and demodulating**

I. INTRODUCTION

Turbo codes proposed by Berrou et. al. [1] represent a recent breakthrough in coding technology, which has stimulated a large amount of related research. The impressive performance of the Turbo codes at low signal-to-noise ratio is due to the structure of the codewords in a concatenation format, the use of pseudorandom interleaving and iterative update of soft information between the concatenated decoders. Turbo codes are concatenated in parallel. However, code concatenation need not always be parallel. Serial concatenation and hybrid combinations may also be possibilities. Concepts of iterative decoding for different concatenated coding schemes have been studied in [2] and [4].

The unprecedented performance of Turbo codes has inspired researchers to consider the application of turbo decoding principle in DS-CDMA mobile radio systems for short frame transmission [7][9][10][11]. Good performances achieved thus far are for the coherent case [9][10]. In this paper, we will consider a DS-CDMA system like the IS-95(A) in which coherent detection is not feasible on the uplink due to the lack of a pilot signal from each user. One way to reduce the effect of interference with manageable system complexity is by using M-ary orthogonal signaling. It can be implemented by using Walsh Hadamard (WH) orthogonal symbols. Similar studies on Turbo coding coupled with WH modulation can be found in [7][11][12]. In [11], Turbo codes was applied to the uplink of a DS-CDMA system mainly for data (large frame) transmission. In [12], a system consisting of rate 1/5 Turbo codes coupled with 32-ary orthogonal modulation signaling was introduced, but the puncturing scheme adopted resulted in poor system performance. In [7], Turbo decoding principle is applied to IS-95(A) uplink with convolutional code and WH modulation.

In this paper, we combine Turbo codes with orthogonal modulation into a hybrid concatenated coding scheme with the parallel-concatenated Turbo coding as outer coding and WH modulation as inner code. This approach is valid because the M-ary WH modulation can be treated as a systematic block code. By modifying the WH demodulator as a soft-in-soft-out (SISO) decoder of the WH symbols, we are able to combine demodulation and decoding functions into one iterative decoder module at the receiver. Performance of this scheme is then thoroughly investigated under various practical considerations, such as short frame transmission for voice traffic, frequency-selective fading channels with different fade rates, and closed-loop power controlled channels with different feedback delays.

In the sequel, the structure of the transmission system under investigation and the channel model for a multiuser DS-CDMA system with RAKE combining are first described in Section II and III. In Section IV, the SISO demodulators for coherent and noncoherent reception of WH modulation are presented. The iterative decoding configuration for the hybrid concatenated coding scheme is then briefly described in Section V. System performances over AWGN and multipath Rayleigh fading channels are finally shown and compared with those obtained by separate decoding of Turbo codes with orthogonal modulation, and iterative decoding and demodulation of covolutional code with orthogonal modulation in Section VI.

II. SYSTEM MODEL

Fig. 1 shows a DS-CDMA transmission system which employs Turbo codes coupled with orthogonal modulation. To add a touch of realism, parameters similar to that of IS-95(A) CDMA uplink are used. The information bits with frame size=192 and bit rate $R_b = 9.6kbps$ are first encoded by a rate $1/3$ constraint length $K=5$ Turbo codes consisting of two identical constituent recursive systematic convolutional (RSC) codes with code generators 31 and 27(in octal). After Turbo encoding, the code symbols are interleaved by a 32×18 block interleaver and then modulated by a 64-ary orthogonal Walsh-Hadamard modulator. The final signal processing elements perform the BPSK modulation and quadrature-phase DS-spreading with a processing gain $N_c = 4$.

The WH modulation may be treated as a systematic block code with rate 6/64 and distance 32. Thus the successive implementation of Turbo codes and orthogonal WH modulation may in itself be viewed as a form of serial code concatenation. Since Turbo codes is already a parallel-concatenated code, the coding and modulation operations can be considered as a form of hybrid concatenation.

III. CHANNEL MODEL

Here we consider an asynchronous K-user DS-CDMA system in a L-path Rayleigh fading channel. A J-branch RAKE receiver is used to offer path or delay diversity reception. The DS-spreading employed in DS-CDMA systems allows the receiver to search the channel for separable multipath components and perform separate despreading of each multipath component. Under the assumption that the time delay of each path τ_l^k can be perfectly estimated, we obtain in Fig. 2 an equivalent non-spread spectrum channel model for the k th user, for $k = 1,\cdots,K$.

First of all, we assume that there are J diversity channels, carrying the same information-bearing signal. Each is assumed to be frequency-nonselective and slowly fading with Rayleigh-distributed envelope $\alpha_j^k (j = 1,\cdots J)$ and uniformly distributed phase shift θ_j^k. The fading processes in the J diversity channels are assumed to be mutually statistically independent with average energy $\Omega_j^k = \mathrm{E}\{(\alpha_j^k)^2\}$. The signal in each channel is then corrupted by an additive noise process. The noise processes in the J channels are assumed to be white Gaussian, mutually independent, with zero-mean and identical variance given by [13]

$$\sigma_T^2 = \frac{N_0}{2} + \sigma_{si}^2 + \sigma_{mai}^2 = \frac{N_0}{2} + \frac{1}{3N_c}\left\{\sum_{l=1,l\neq j}^{L} \Omega_j^k + \sum_{i=1,i\neq k}^{K} \sum_{l=1}^{L} \Omega_l^i \right\} \tag{1}$$

By the term σ_T^2, we take into account the influence of thermal noise, multiple access interference (MAI) and RAKE-path interference as a result of non-ideal orthogonality properties of the PN-code. For orthogonal demodulation, M Walsh Hadamard matched filters or one equivalent Fast Hadamard Transform (FHT) matched filter are incorporated into each parallel branch. The output of each branch is then a complex correlation vector of length M. They are combined by the diversity combiner to form a real vector $W = \{w_j\}$, for $j = 0,\ldots,M-1$. For coherent reception, we use maximum ratio combining (MRC) technique; while for noncoherent detection, we use square law combining (SLC). The output of the diversity combiner is finally fed into the channel decoder.

IV. INNER WALSH HADAMARD MODULATION

An M-ary WH symbol can be viewed as a $\left(m, M, \frac{M}{2}\right)$ systematic block code with information position set $\mathbf{I}_m = \{1,2,4,\cdots,2^{m-1}\}$, where $m = \log_2 M$. It is obtained by selecting as codewords the rows $\mathbf{h}_j$ of a $M \times M$ Hadamard matrix $\mathbf{H}_m$. The Hadamard matrix is generated from $\mathbf{H}_{m-1}$ according to the recursive relation

$$\mathbf{H}_m = \begin{bmatrix} \mathbf{H}_{m-1} & \mathbf{H}_{m-1} \\ \mathbf{H}_{m-1} & -\mathbf{H}_{m-1} \end{bmatrix} \tag{2}$$

starting with a 1×1 unity matrix $\mathbf{H}_0 = [1]$.

Conventional WH demodulation as that in the IS-95(A) standard was realized by making a symbol decision based on the maximum element of the vector $\mathbf{W}$ produced from the diversity combiner. The transmitted binary data is obtained by symbol-to-bit demapping and the log-likelihood function of the corresponding element is then used as a soft decision metric in the channel decoding. In order to combine the demodulating and decoding into one iterative decoder module, we modify the WH demodulation as the decoding of WH symbols with a soft-in-soft-out (SISO) decoder module, which accepts soft a-priori information and make use of all the elements of $\mathbf{W}$ to recalculate its soft decision by exploiting the memory inherent in this modulation scheme. With such a SISO demodulator, we are then able to exchange bitwise, instead of symbolwise, soft decision through feed forward and feedback loops between the inner WH decoder and outer Turbo decoder.

To derive the a-posteriori log-likelihood ratio (LLR) for the i th $(i \in \mathbf{I}_m)$ information bit, we use the symbol-by-symbol MAP algorithm for block code [2], that is,

$$\mathrm{L}(\hat{u}_i) = \ln \frac{\sum\limits_{\mathbf{h}_j \in \mathbf{H}_m : u_i = +1} \mathrm{P}(\mathbf{h}_j | \mathbf{W})}{\sum\limits_{\mathbf{h}_j \in \mathbf{H}_m : u_i = -1} \mathrm{P}(\mathbf{h}_j | \mathbf{W})} = \ln \frac{\sum\limits_{j=0, u_i=+1}^{M-1} p(\mathbf{W} | \mathbf{h}_j) \mathrm{P}(\mathbf{h}_j)}{\sum\limits_{j=0, u_i=-1}^{M-1} p(\mathbf{W} | \mathbf{h}_j) \mathrm{P}(\mathbf{h}_j)} \tag{3}$$

where $p(\mathbf{W}|\mathbf{h}_j)$ is the channel measurement based on the output of the diversity combiner, and $P(\mathbf{h}_j)$ is the a-priori probability of codeword $\mathbf{h}_j$, which can be easily converted from the a-priori LLRs of corresponding input information bits. For $p(\mathbf{W}|\mathbf{h}_j)$, consider the following two cases:

A. coherent detection

The realization of coherent detection is based on the assumption that the channel attenuation and phase shifts are known perfectly. The outputs of each parallel branch can be summed up using MRC technique. It can be shown in [14] that when the desired signal is present, the conditional probability density function (pdf) of the resulting vector element w_j is $p_n(w_j)$, while the pdf when the signal is not present is $p_c(w_j)$.

$$p(w_j | \mathbf{h}_i) = \begin{cases} p_n(w_j) = \dfrac{1}{\sqrt{2\pi\gamma}\sigma_T} \exp\left(-\dfrac{(w_j - \gamma)^2}{2\sigma_T^2 \gamma} \right), & i = j \\ p_c(w_j) = \dfrac{1}{\sqrt{2\pi\gamma}\sigma_T} \exp\left(-\dfrac{w_j^2}{2\sigma_T^2 \gamma} \right), & i \neq j \end{cases} \tag{4}$$

where $\gamma = \sum_{i=1}^{J} (\alpha_i)^2$, and α_i is the fade amplitude of path i.

B. noncoherent detection

In noncoherent detection, estimation of the phase distortion factor due to multipath is not attempted. As long as the phase distortion is nearly constant or at least slowly varying over a symbol duration, SLC rule may be used for diversity combining. From [13],

$$p(w_j | \mathbf{h}_i) = \begin{cases} p_n(w_j) = \dfrac{1}{2\sigma_T^2} \left(\dfrac{w_j}{\gamma} \right)^{\frac{J-1}{2}} \exp\left(-\dfrac{w_j + \gamma}{2\sigma_T^2} \right) \mathrm{I}_{J-1}\left(\dfrac{\sqrt{w_j \gamma}}{\sigma_T^2} \right) \\ p_c(w_j) = \dfrac{1}{(2\sigma_T^2)^J (J-1)!} w_j^{J-1} \exp\left(-\dfrac{w_j}{2\sigma_T^2} \right) \end{cases} \tag{5}$$

where $I_{J-1}(\cdot)$ is the modified Bessel function of first kind and order $J-1$.

Now we have:

$$p(\mathbf{W}|\mathbf{h}_i) = p_c(w_0)\cdots p_n(w_i)\cdots p_c(w_{M-1}) = \prod_{j=0}^{M-1} p_c(w_j)\frac{p_n(w_i)}{p_c(w_i)} = C_1\frac{p_n(w_i)}{p_c(w_i)}$$

$$= \begin{cases} C_2 \exp\left(\dfrac{w_i}{\sigma_T^2}\right), & \text{coherent} \\ C_3 w_i^{-\frac{J-1}{2}} \mathrm{I}_{J-1}\left(\dfrac{\sqrt{w_i\gamma}}{\sigma_T^2}\right) & \text{noncoherent} \end{cases} \tag{6}$$

where $C_2 = C_1\exp\left(-\dfrac{\gamma}{2\sigma_T^2}\right)$, and $C_3 = C_1(J-1)!\left(\dfrac{2\sigma_T^2}{\sqrt{\gamma}}\right)^{J-1}\exp\left(-\dfrac{\gamma}{2\sigma_T^2}\right)$.

To reduce the computational cost, efficient techniques to evaluate Equation (3) as proposed in Appendix A and B of [3] can be adopted. Firstly, the conversion of LLR of information bits to the probability $P(\mathbf{h}_j)$ of a WH symbol can be realized with 2^m additions by the proposed reduced-input-FHT algorithm. The summations in the numerator and denominator of Equation (3) can also be efficiently evaluated by an algorithm similar to the reduced-output-MAP-FHT described in [3] but with some modifications as follows (on Equation (5a) and (5b) in [3])

$$\mathbf{a}[j] = \begin{bmatrix} \exp\left(\dfrac{\mathbf{h}_j\cdot\mathbf{L}}{2}\right) \\ 0 \end{bmatrix}, \qquad \mathbf{b}[j] = \begin{bmatrix} \sum\limits_{\mathrm{H}[i,j]=+1} \exp\left(\dfrac{\mathbf{h}_i\cdot\mathbf{L}}{2}\right) \\ \sum\limits_{\mathrm{H}[i,j]=-1} \exp\left(\dfrac{\mathbf{h}_i\cdot\mathbf{L}}{2}\right) \end{bmatrix} \tag{7}$$

Thus all the multiplications in Equation (3) are simplified to additions and the implementation complexity is drastically reduced to $3\cdot 2^m$ additions. The total operation for a i-iteration SISO WH demodulator is $i\cdot N\cdot 2^{m+2}$ for N WH symbols.

V. ITERATIVE DECODING CONFIGURATION

Although the SISO Inner WH demodulators are somewhat different in coherent and noncoherent case, the constituent SISO1/SISO2 decoder module of the outer Turbo decoder are the same. They may be constructed by using BCJR-MAP, SOVA, or their variants [2]. In this simulation, we chose BCJR-MAP.

Fig. 3 shows the decoding configuration for the resultant hybrid concatenated coding scheme. All the SISO module formats take the definition of [5]. The SISO Inner demodulator acts in the same way as the inner decoder of a serially concatenated system. The two constituent Turbo decoders calculate its branch metrics by using the soft information provided by the WH demodulator, and the extrinsic LLR from the other constituent decoder as they would do in a parallel concatenated coding scheme. The difference here is that for the Turbo decoder, trace back is performed not only for its *information* bits but also for its *coded* bits. This configuration operates in a serial mode, that is, SISO Inner processes before SISO1 and SISO1 processes before SISO2. It may not be the optimal operation mode but we adopt it for the sake of implementation simplicity.

VI. SIMULATION RESULTS AND DISCUSSION

For comparison purposes, we have also simulated Turbo codes (TC) and WH modulation without combined decoding and the combined iterative decoding of convolutional code (CC) and WH modulation proposed by Hagenauer [7] for the IS-95(A) uplink. These two approaches are denoted as *tc_wh separate decoding* and *cc_wh combined decoding*, respectively. The combined decoding of Turbo codes and WH modulation proposed herein is denoted by *tc_wh combined decoding*. Twelve iterations for each frame are performed before final decision is made. Notice that N_0 in the figures is defined as the sum of thermal noise and MAI, i.e., $\frac{N_0}{2}+\sigma_{mai}^2$ of Equation (1).

Fig. 4 and Fig. 5 illustrate the bit error rate (BER) performance of the three concatenated coding approaches over coherent and noncoherent AWGN channels, respectively. The results show that the gains of coherent reception over noncoherent reception for each approach are about 1.5dB (BER=10^{-3}).

As shown in Fig. 4 and 5, the coding gain of *tc_wh combined decoding* over *tc_wh separate decoding* is considerable. At the BER level of 10^{-3}, it is 0.7-0.8dB. On the other hand, the gain of *tc_wh combined decoding* over *cc_wh combined decoding* is 0.3-0.5dB. The slopes of the BER curves (BER roll-off rates) also seem to be steeper for *tc_wh* schemes than the *cc_wh* schemes, especially in the noncoherent case (Fig. 5). This is believed to be the consequence of better code distance property of the former. In the noncoherent AWGN channel, we also reproduce the simulation result (*A1: BER*) from [6] for the current IS-95(A) uplink performance in AWGN channel without iterative decoding. At the BER of 10^{-3}, the gain of *tc_wh combined decoding* over it is approximately 2dB, of which 1dB is obtained by ideal soft decoding (denoted by *cc_wh separate decoding*) and the reminding 1dB from iterative decoding and demodulating of the combined Turbo codes and orthogonal modulation.

It should be noted that the decoding complexity of the outer Turbo decoder is approximately the same as that of a single Turbo decoder, since after the LLR-calculation for its *information* bits, the LLRs for its *coded* bits are readily available. The extra computational cost of *tc_wh combined decoding* over *tc_wh separate decoding* thus comes from the SISO Inner demodulator, which is shown in Section IV to be $i \cdot N \cdot 2^{m+2}$ addition operations. Because the complexity of a Turbo decoder (with $K=5$ in this study) is much less than that of a larger-constraint-length convolutional decoder (with $K=9$ for IS-95(A)), the combined Turbo codes and orthogonal modulation approach is capable of offering not just performance, but also cost benefits.

The frame error rate (FER) performances of the two decoding schemes of Turbo codes and WH modulation, i.e. *tc_wh separate decoding* and *tc_wh combined decoding*, in 2-path Rayleigh fading channel have also been investigated. In the simulation results shown in Fig. 6, each path has the same signal strength and experiences independent fading. There are six groups of curves. The rightmost group obtained with a mobile speed of 10 km/h without RAKE reception gives the worst performance, while the leftmost group achieved in random fading channel with 2-tap RAKE receiver gives the best performance. All results in

Fig. 6 show that *combined decoding* consistently outperforms *separate decoding* by about 1dB at FER=10^{-2}.

Fig. 6 is obtained without power control. Further results for TC_WH with closed-loop power control (CL-PC) [8] are shown in Fig. 7. 1dB power adjustment step size, 6-WH-symbol (1.25ms) update interval and no bit-stream command error are assumed. To investigate the effect of PC feedback delay on performance, two values are simulated: 1.25ms (denoted as *pc1*) and 2.5ms (denoted as *pc2*). It can be observed that performance is more sensitive to the feedback delay in fast fading (v=50km/h) than in slow fading (v=10km/h) channels. This is because PC is able to keep track of the amplitude fluctuation at slow fading rate. With RAKE reception, this sensitivity is enhanced since the diversity-combined signal suffers less fades and is more easily tracked. This shows that the effectiveness of PC is significantly aided by RAKE combining.

CL-PC, RAKE diversity receiver and channel interleaving are used in practical CDMA systems to combat multipath fading. To better illustrate their effects, we put together pertinent curves from Fig. 6 and Fig. 7 into Fig. 8. It's interesting to see that without RAKE (dashed curves), the performance in power-controlled fast fading channel is better than the performance in random fading channel without power control in the high range of FER ($\geq 10^{-2}$) and the curves cross over at FER=10^{-2}. However, with 2-tap RAKE (solid curves), the performance in random fading is always better than those with CL-PC by around 2dB. This suggests that highest system capacity can be obtained by randomizing the fading channel instead of CL-PC (since the tracking capacity of CL-PC is inherently limited by its update frequency and the channel fading rate). Random fading for short frame transmission can be approached by a combination of higher carrier frequency and deterministic carrier hopping, in addition to the conventional time-interleaving.

The result (*A2(FER)*) reported in [6] for conventional IS-95(A) uplink performance in 2-path random Rayleigh fading channel was also reproduced in Fig. 8. The proposed scheme in this paper demonstrates a better performance of 2dB at the FER level of 10^{-2}.

VII. CONCLUSION

We study a DS-CDMA system which employs combined iterative decoding of Turbo codes and orthogonal modulation for short frame transmission. It is shown that considerable performacne improvement can be obtained with moderate computational efforts and the best system performance is obtained with RAKE reception and randomizied Rayleigh fading (instead of CLPC). In this paper, the interleaver used between Turbo codes and orthogonal modulation is a block interleaver. Performance with random interleavers in between the constituent coders (code interleaver) as well as before the fading channel (channel interleaver) are currently under investigation. But now extra delay is introduced into the DS-CDMA system by the channel interleaver. It may not be acceptable in mobile radio communications for voice transmission. Further studies on the trade-off between performance and delay is hence instructive.

VIII. REFERENCE

[1] C. Berrou, A. Glavieux and P. Thitimajshima, "Near Shannon limit error correcting coding and decoding: turbo-codes," Proc. ICC93, pp. 1064-1070.

[2] Joachim Hagenauer, "Iterative Decoding of Binary Block and Convolutional Codes," IEEE Tran. Inf. Theory, vol. 42, no. 2, pp. 429-445, Mar. 1996.
[3] Li Ping and Sammy Chan, "Iterative Decoding of Concatenated Hadamard Codes," IEEE Conf. Commun., vol. 1, pp. 136-140, 1998.
[4] D. Divsalar, and F. Pollara, "Serial and Hybrid Concatenated Codes with Applications," IEEE Proc. International Symposium Inf. Theory, pp. 10, 1997.
[5] S. Benedetto, D. Divsalar, G. Montorsi, and F. Pollara, "A Soft-Input Soft-Output APP Module for Iterative Decoding of Concatenated Codes," IEEE Commun. Letters, vol. 1, no. 1, pp. 22-24, Jan 1997.
[6] Li Fung Chang, Fuyun Ling, David D. Falconer, and Nelson R. Sollenberger, "Comparison of Two Convolutional Orthogonal Coding Techniques for CDMA Radio Communications Systems," IEEE Trans. Commun., vol. 43, no. 6, pp. 2028-2036, Jun. 1995.
[7] Rupert Herzog, Andreas Schmidbauer and Joachim Hagenauer, "Iterative Decoding and Despreading Improves CDMA-Systems Using M-ary Orthogonal Modulation and FEC," IEEE International Conf. Commun., vol. 2, pp. 909-913, 1997.
[8] Andrew J. Viterbi, Audrey M. Viterbi and Ephraim Zehavi, "Performance of Power-Controlled Wideband Rerrestrial Digital Communication," IEEE Trans. Commun., vol. 41, no. 4, pp. 559-560, Apr. 1993
[9] Peter Jung, "Comparison of Turbo-Code Decoders Applied to Short Frame Transmission Systems," IEEE J. Select. Areas Commun., vol. 14, no. 3, pp530-537, Apr. 1996.
[10] Vladislav Sorokine, Frank R. Kschischang, Subbarayan Pasupathy, "Gallager Codes for CDMA Applications: Generalizations, Constructions and Performances," Inf. Theory Workshop, pp. 8 –9, 1998.
[11] Bruno Melis, and Giovanni Romano, "Application of Turbo codes in the uplink of a DS-CDMA system," Proc. International Symposium on Turbo codes, pp. 243-246, Brest, France, 1997.
[12] Naftali Chayat, "Turbo Codes for Incoherent M-ary Orthogonal Signaling," IEEE Convention Electrical & Electronic Engineering in Israel, pp. 471-474, 1996.
[13] V. Aalo, O. Ugweje, and R. Sudhakarm "Performance Analysis of a DS/CDMA with Noncoherent M-ary Orthogonal Modulation in Nakagami Fading," IEEE Trans. Veh. Tech., vol. 47, pp. 20-29, Feb. 1998.
[14] Proakis, "Digital Communications," New York McGraw-Hill.

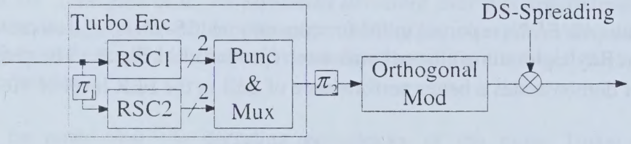


Fig. 1 Turbo codes with orthogonal modulation in a DS-CDMA system

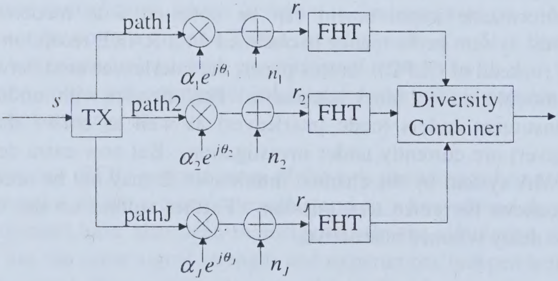


Fig. 2 Equivalent channel model for an asychronous multiuser DS-CDMA system employing J -branch RAKE receiver, where $n_j, \alpha_j, \theta_j\,(j = 1,..., J)$ denotes the additive noise process, path gain and phase shift of the j th multipath, respectively

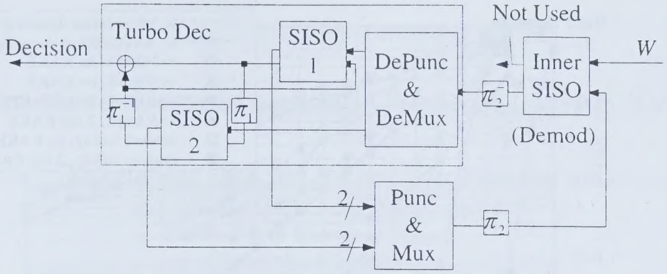

Not Used
Decision
Turbo Dec
SISO
1
DePunc
&
DeMux
W
Inner
SISO
π_2^{-1}
π_1^{-1}
SISO
2
π_1
(Demod)
2
Punc
&
Mux
π_2
2

Fig 3. Iterative decoding configuration for the combined Turbo codes with WH modulation

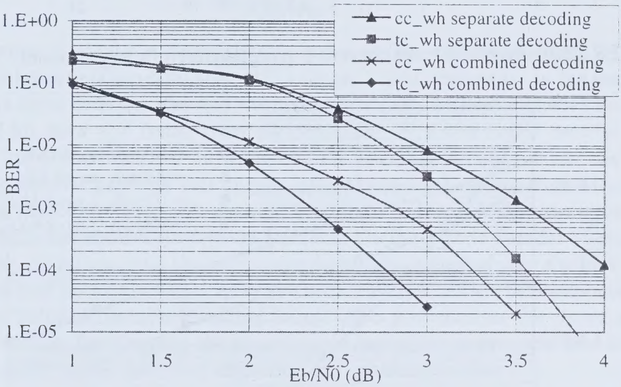


Fig 4. BER performance in coherent AWGN channel

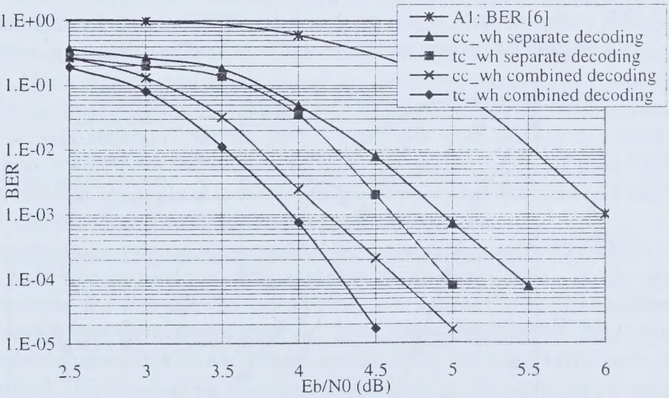


Fig. 5 BER performance in noncoherent AWGN channel

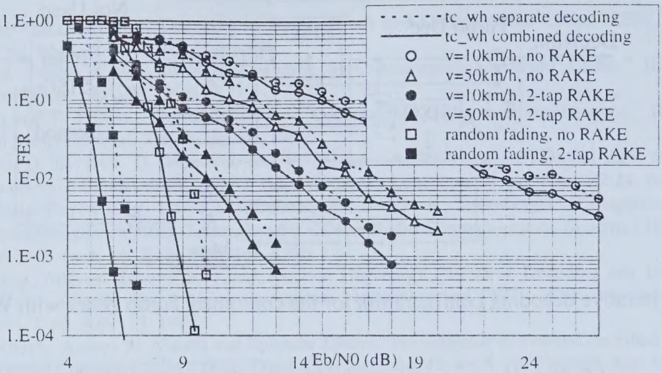


Fig. 6 FER performance in noncoherent 2-path Rayleigh fading channel, $f_c = 900\text{MHz}$

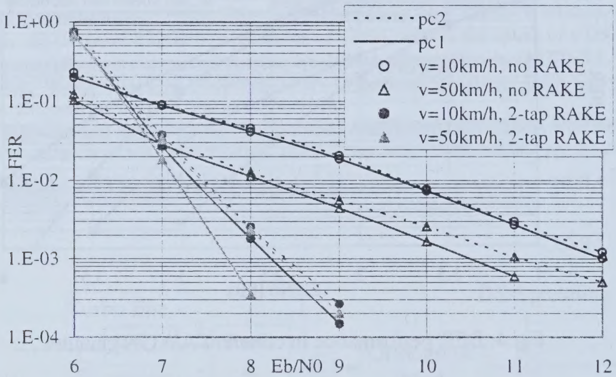


Fig. 7 FER performance of *tc_wh combined decoding* in 2-path Rayleigh fading channel with closed-loop power control of different feedback delays, $f_c = 900\text{MHz}$

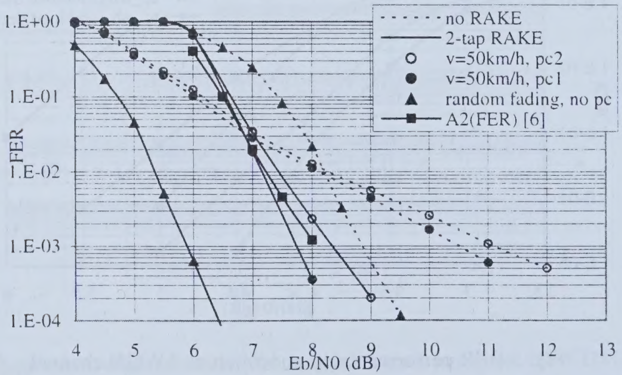


Fig. 8 FER performance of *tc_wh combined decoding* in 2-path Rayleigh fading channel

An Interactive Concatenated Turbo Coding System*

Ye Liu, Heng Tang, Shu Lin and Marc Fossorier
Department of Electrical Engineering
University of Hawaii at Manoa
Honolulu, HI 96822
yliu@spectra.eng.hawaii.edu

Abstract

This paper presents a concatenated turbo coding system in which a Reed-Solomon outer code is concatenated with a binary turbo inner code. In the proposed system, the outer code decoder and the inner turbo code decoder interact to achieve both good bit error and frame error performances. The outer code decoder helps the inner turbo code decoder to terminate its decoding iteration while the inner turbo code decoder provides soft-output information to the outer code decoder to carry out a reliability-based soft-decision decoding. In the case that the outer code decoding fails, the outer code decoder instructs the inner code decoder to continue its decoding iterations until the outer code decoding is successful or a preset maximum number of decoding iterations is reached. This interaction between outer and inner code decoders reduces decoding delay. Also presented in the paper are an effective criterion for stopping the iteration process of the inner code decoder and a new reliability-based decoding algorithm for nonbinary codes.

1. Introduction

Although turbo codes with iterative decoding [1,2,3] have been shown to achieve bit-error rates (BER's) of 10^{-5} or better at SNR's within 1 dB of the SNR for which the code rate equals channel capacity, they suffer from three disadvantages: (1) a large decoding delay due to the large block lengths and many decoding iterations required for near capacity performance, (2) an error floor at low BER's due to the fact that the component codes have relatively poor minimum distances, which manifests itself at very low BER's, and (3) a relatively poor frame error performance (FER). The error floor and poor FER make turbo codes not suitable for applications requiring extremely low BER's and FER's, such as some control command and compressed package applications.

This paper presents an *interactive concatenated turbo coding system* in which a Reed-Solomon outer code is concatenated with a binary turbo inner code. In the proposed system, the outer code decoder and the inner turbo code decoder *interact* to achieve both good bit error and frame error performances. The outer code decoder helps the inner turbo code decoder to terminate its decoding iteration while the inner turbo code decoder provides soft-output information to the outer code decoder to carry out a reliability-based soft-decision decoding. In the case that the outer code decoding fails, the outer code decoder instructs the inner code decoder to continue its decoding iterations until

---

[1] This research was supported by NSF under Grants NCR 94-15374, CCR 97-32959, CCR 98-14054 and NASA under Grant NAG 5-931.

the outer code decoding is successful or a preset maximum number of decoding iterations is reached. This interaction between outer and inner code decoders reduces decoding delay. Also presented in the paper are an effective criterion for stopping the iteration process of the inner code decoder and a new reliability-based decoding algorithm called *Chase-GMD algorithm* for nonbinary codes.

2. Parallel Turbo Decoding and BM Stopping Criterion

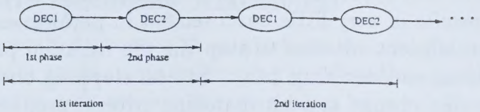
A turbo code with linear block codes as component codes is obtained by parallel concatenation of two systematic linear block codes with a pseudo random interlever between two encoders. Assume that two component codes are identical and both are (n_i, k_i) binary linear block codes. Let $\mathbf{u} = (u_0, u_1, \cdots, u_{K-1})$ be the information sequence to be encoded for transmission where $K = lk_i$. The first encoder encodes this sequence and produces a block of $l(n_i - k_i)$ parity-check bits, denoted $\mathbf{p}^{(1)}$. The interlever permutes the information sequence $\mathbf{u}$ into a sequence $\mathbf{u}' = \Pi(\mathbf{u})$. The second encoder encodes $\mathbf{u}'$ and produces a block of $l(n_i - k_i)$ parity-check bits, denoted $\mathbf{p}^{(2)}$. Then the sequence $(\mathbf{u}, \mathbf{p}^{(1)}, \mathbf{p}^{(2)})$ is the code sequence for the information sequence $\mathbf{u}$. The collection of 2^K such code sequences, one for each information sequence $\mathbf{u}$, form a turbo code of length $N = l(2n_i - k_i)$. Since the component codes are block codes, it is called a block turbo code.

The decoder for a turbo code with two component codes consists of two soft-input/soft-output (SISO) MAP (or APP) decoders which operate iteratively [1,2,3]. Since the two component codes are identical, the two MAP decoders are identical. Decoding can be carried out in either serial mode or parallel mode as shown in Figures 1(a) and 1(b), respectively. The serial decoding mode was originally proposed by Berrou et. al. [1], and the parallel decoding mode was later proposed by Divsalar and Pollara [6]. In serial mode, decoder 1 and decoder 2 , denoted DEC1 and DEC2, respectively, operates alternately. In parallel mode, the two MAP decoders, DEC1 and DEC2, operate simultaneously. Decoding consists of a sequence of iterations, each decoding iteration consists of two phases. In serial mode, DEC1 operates in the first phase and DEC2 operates in the second phase as shown in Figure 1(a). However in parallel mode, both DEC1 and DEC2 operates in each phase as shown in Figure 1(b).

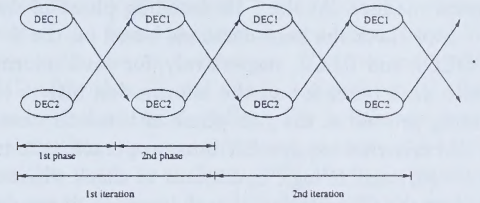
In the proposed interactive concatenated turbo coding system, inner turbo decoding is performed in parallel mode. Suppose a code sequence $(\mathbf{u}, \mathbf{p}^{(1)}, \mathbf{p}^{(2)})$ is transmitted. Let $\mathbf{r} = (r_0, r_1, \cdots, r_{N-1})$ be the received sequence. For decoding, this received sequence is decomposed into two subsequences $\mathbf{r}^{(1)} = (r_0^{(1)}, r_1^{(1)}, \cdots)$ and $\mathbf{r}^{(2)} = (r_0^{(2)}, r_1^{(2)}, \cdots)$, corresponding to code sequences $(\mathbf{u}, \mathbf{p}^{(1)})$ and $(\mathbf{u}', \mathbf{p}^{(2)})$ at the outputs of two component encoders, respectively. Each SISO decoder has two inputs and two outputs. The inputs to each decoder are the a priori L-values (log-likelihood values) $L(u_i)$ for all information bits u_i and the received channel L-values $L_c r_i$ for all code bits, where $L_c = 4aE_s/N_0$ and E_s/N_0 is the channel SNR. For a fading channel, a denotes the fading amplitude, whereas for an AWGN channel, $a = 1$. Based on its inputs, the SISO decoder computes L-values (soft outputs)

$$L(\hat{u}_i) \triangleq L(u_i|\mathbf{r}) = \log \frac{p(u_i = 1|\mathbf{r})}{p(u_i = 0|\mathbf{r})},$$

for all information bits and delivers an extrinsic L-value $L_e(\hat{u}_i)$ for each information bit which contains the reliability information from all other coded bits in the code sequence and is not influenced by $L(u_i)$ and $L_c r_i$ of the current bit u_i.



(a) Serial



(b) Parallel

Figure 1: Serial and parallel decoding structures of turbo codes

Consider turbo decoding in parallel mode. At the first iteration, DEC1 and DEC2 start decoding at the same time. The inputs to DEC1 and DEC2 are channel L-values $L_c r_i^{(1)}$ and $L_c r_i^{(2)}$, respectively. For equally likely information bits, the a priori L-value $L(u_i)$ inputs to both SISO decoders in the first phase of the first iteration are zero. Hence, we set $L^{(1)}(u_i) = 0$ and $L^{(2)}(u_i) = 0$ for each information bit. The outputs of DEC1 and DEC2 are L-values $L^{(1)}(\hat{u}_i)$ and $L^{(2)}(\hat{u}_i)$ and extrinsic L-values $L_e^{(1)}(\hat{u}_i)$ and $L_e^{(2)}(\hat{u}_i)$, respectively, with $0 \leq i < K$. Then the second phase starts. The inputs to DEC1 are channel L-values $L_c\mathrm{r}^{(1)}$ and extrinsic values $L_e^{(2)}(\hat{u}_i)$'s which are the outputs of DEC2 in the first decoding phase. The inputs to DEC2 are $L_c\mathrm{r}^{(2)}$ and $L_e^{(1)}(\hat{u}_i)$'s which are the outputs of DEC1 in the first decoding phase. The second decoding phase is then performed. All the subsequent iterations are carried out in the same manner as the first iteration except that the a priori L-values $L^{(1)}(u_i)$ and $L^{(2)}(u_i)$ of each information bit to the inputs of the two SISO decoders at the first decoding phase are the extrinsic L-values $L_e^{(2)}(\hat{u}_i)$ and $L_e^{(1)}(\hat{u}_i)$, respectively, which are the outputs of the two decoders in the second decoding phase of the previous iteration.

After a sufficient number of iterations (or decoding phases), we can stop the decoding process and obtain the L-value for each information bit as follow:

$$L(\hat{u}_i) = \begin{cases} L^{(1)}(\hat{u}_i), & \text{if } |L^{(1)}(\hat{u}_i)| \geq |L^{(2)}(\hat{u}_i)|, \\ L^{(2)}(\hat{u}_i), & \text{otherwise,} \end{cases} \tag{1}$$

where

$$L^{(1)}(\hat{u}_i) = L_c \cdot r_i^{(1)} + L^{(1)}(u_i) + L_e^{(1)}(\hat{u}_i), \text{ and } L^{(2)}(\hat{u}_i) = L_c \cdot r_i^{(2)} + L^{(2)}(u_i) + L_e^{(2)}(\hat{u}_i).$$

Finally, the hard-decision decoded information bit u_i is made based on

$$\hat{u}_i = \begin{cases} 0, & \text{if } L(\hat{u}_i) \leq 0 \\ 1, & \text{if } L(\hat{u}_i) > 0, \end{cases}$$

for $0 \leq i < K$.

As the iterative decoding approaches the performance limit of a given turbo code, any further iteration results in very little improvement in performance. Therefore it is important to devise an efficient criterion to stop the the iteration process and prevent unnecessary computations and decoding delay. Several stopping criteria have been devised [3,7]. Both the sign change and bit matching criteria proposed in [7] are more computationally efficient than the cross entropy (CE) criterion proposed in [3].

The bit matching (BM) criterion of [7] can be applied to terminate decoding in parallel mode in a straightforward manner. At the j-th decoding phase of the k-th iteration for $j = 1, 2$ and $k = 1, 2, \cdots$, we check the hard decisions based on the L-values $L^{(1)}(\hat{u}_i)$ and $L^{(2)}(\hat{u}_i)$ generated by DEC1 and DEC2, respectively, for each information bit. If these hard decisions agree with each other for all the information bits in the whole sequence, we terminate the decoding process at the j-th phase of the k-th iteration.

At each phase, the BM criterion requires $2K$ binary operations to make hard decisions based on $L^{(1)}(\hat{u}_i)$ and $L^{(2)}(\hat{u}_i)$ and K logic operations to check whether the BM criterion is satisfied. However to test the CE criterion at each iteration, it requires a total of $5K-1$ real number operations, including $2K-1$ additions and subtractions, $2K$ multiplications and divisions, and K exponentiations. Therefore, the BM criterion requires much simpler computations than the CE criterion.

Simulation results show that the BM criterion saves more iterations than the CE criterion with negligible degradation in error performance. Consider the turbo code with the (64,57) distance-4 extended Hamming code as the two component codes and a block interleaver of size $K = 57 \times 57$. The error performances of decoding in parallel mode with BM stopping criterion and serial mode with CE stopping criteria are shown in Figure 2. We see that decoding in parallel mode with BM stopping criterion outperforms decoding in serial mode with CE stopping criterion. The average numbers of decoding iterations required using BM and CE criteria, respectively, for parallel mode decoding of the above turbo code are also shown in Figure 2. We see that the BM stopping criterion saves more decoding iterations than the CE stopping criterion and hence reduces computational complexity. From Figure 2, we also see that the frame error performance is relatively poor compared with the bit error performance. The error floor starts at frame error probability of 10^{-2}. This error floor will be removed when the proposed concatenated turbo system is used.

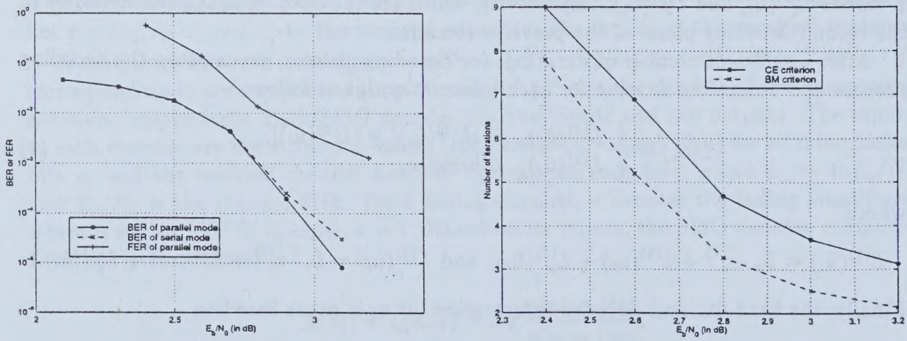


Figure 2: Error performance and number of iterations of turbo decodings of the (64,57) Hamming code

3. Chase-GMD Decoding Algorithm

RS codes are commonly decoded with an algebraic decoding algorithm, such as the Euclidean Algorithm, in applications for keeping the decoding complexity low. To improve the error performance, soft-decision decoding must be used. However, soft-decision decoding of RS codes significantly increases the decoding complexity. One approach to improve the performance of algebraic decoding while keeping low decoding complexity is to use an algebraic decoder to generate a sequence of candidate codewords based on the reliability values of the received symbols, and then choose the candidate codeword with the best metric as the decoded codeword. The two most well known such decoding algorithms are GMD algorithm [5] and Chase-2 algorithm [4]. Both algorithms improve the error performance of algebraic decoding. For a RS code over GF(q) with minimum distance d. GMD algorithm requires to perform at most $\lfloor (d+1)/2 \rfloor$ algebraic decodings while Chase-2 algorithm needs to perform $q^{\lfloor d/2 \rfloor}$ algebraic decodings based on $q^{\lfloor d/2 \rfloor}$ test error patterns with errors confined to $\lfloor d/2 \rfloor$ least reliable positions of the received sequence. Chase-2 algorithm outperforms GMD algorithm, however it requires much more decoding computations. For long RS codes over large field GF(q) with large minimum distance d, Chase-2 algorithm becomes impractical. The GMD decoding algorithm while simple gives only small improvement in error performance over pure algebraic decoding for small to medium SNR's, especially for long RS codes. Therefore, GMD is not very attractive for practical applications for small to medium SNR's and therefore, it must be improved.

In this section, we present a decoding algorithm which combines Chase-2 and GMD algorithms. It provides a good trade-off between the error performance of Chase-2 algorithm and the decoding complexity of GMD algorithm. We call this decoding algorithm Chase-GMD algorithm.

Consider an (n_o, k_o, d) RS code over GF(q) with $q = 2^m$. Let $\mathbf{x} = (\mathbf{x}_0, \mathbf{x}_1, \cdots, \mathbf{x}_{n_o-1})$ be a codeword. For binary transmission, every code symbol $\mathbf{x}_i$ is expanded into a binary m-tuple. Let $\mathbf{y} = (\mathbf{y}_0, \mathbf{y}_1, \cdots, \mathbf{y}_{n_o-1})$ be the unquantized received sequence at the output of the matched filter in the receiver, where $\mathbf{y}_i$ represents a vector $(y_{i,0}, y_{i,1}, \cdots, y_{i,m-1})$ composed of m real numbers. Let $\mathbf{z} = (\mathbf{z}_0, \mathbf{z}_1, \cdots, \mathbf{z}_{n_o-1})$ be the hard-decision received sequence obtained from $\mathbf{y}$ with $\mathbf{z}_i$ in GF(2^m). A real number α_i is assigned to each hard-decision received symbol $\mathbf{z}_i$ to indicate its reliability. There are a number of ways to define α_i's [5,8]. For the proposed concatenated turbo coding system, since inner turbo decoding not only gives the hard-decision of each information bit but also provides its reliability L-value. Based on the bit reliability values, we can easily compute the reliability value of each hard-decision received symbol $\mathbf{z}_i$. Let $(x_{i,0}, x_{i,1}, \cdots, x_{i,m-1})$ be the binary m-tuple expansion of code symbol $\mathbf{x}_i$. With respect to inner encoding, each $x_{i,j}$ with $0 \leq j < m$, is an information bit. The inner turbo decoding provides each bit $x_{i,j}$ a reliability L-value, $L(\hat{x}_{i,j})$. Then the reliability value of the i-th hard-decision received symbol $\mathbf{z}_i$ is

$$\alpha_i = \prod_{j=0}^{m-1} p(\hat{x}_{i,j}), \text{ where } p(\hat{x}_{i,j} = \pm 1) = \frac{e^{\pm L(\hat{x}_{i,j})}}{1 + e^{\pm L(\hat{x}_{i,j})}},$$

for $0 \leq i < n_o$ and $0 \leq j < m$. The larger α_i, the more reliable $\mathbf{z}_i$ is.

Now we describe the Chase-GMD algorithm. Without loss of generality, we assume that the hard-decision received symbols in $\mathbf{z}$ are ordered in the order of increasing reliability such that $\alpha_i \leq \alpha_j$ for $i < j$. We also assume that an error-and-erasure algebraic decoder [5] is used to generate candidate codewords which corrects e errors and s erasures provided that $s + 2e < d$. For $0 \leq P \leq \lfloor d/2 \rfloor$, let E denote the set of q^P test error patterns with errors (nonzero components) confined to the P least reliable positions. Let $A_i(q')$ denote the set of $q' < q$ symbols in GF(q) at the i-th symbol position that give the q' largest conditional probabilities $p(\mathrm{y}_i|g)$, $g \in \mathrm{GF}(q)$, $0 \leq i < P$. The error at the i-th position of E is chosen from $A_i(q')$. There are q'^P error patterns in E. Let CGA(P, q') denote the Chase-GMD algorithm with parameter P and q'. This CGA(P, q') processes all the vectors $\mathbf{w} = \mathbf{z} + \mathbf{e}$ with $\mathbf{e}$ in E. Define the following set of integers:

$$I(P) = \{i : \ 0 \leq i \leq d - 2P - 1 \text{ and } d - i \text{ is odd}\}.$$

For each $\mathbf{w}$ and each integer $i \in I(P)$, erase i symbols of $\mathbf{w}$ starting from symbol position $P + 1$ to symbol position $P + i$. This results in a vector $\mathbf{w}^*$ with i erasures. Perform error-and-erasure decoding on $\mathbf{w}^*$. If decoding is successful, the decoded codeword is a candidate codeword. After performing

$$q'^P(\lfloor (d+1)/2 \rfloor - P) \tag{2}$$

decodings, we obtained a set of candidate codewords. Among these candidate codewords, the one with the best metric is the decoded codeword. For $P = 0$, CGA(0,0) is simply the GMD algorithm and for $P = \lfloor d/2 \rfloor$, CGA(P, q) is simply the Chase-2 algorithm. It can be proved that the performance of CGA(P) improves as P increases (the proof is omitted here).

Consider the decoding (31,25,7) RS code over GF(2^5). Then $q = 32$. Suppose we chose $q' = 2$. For $P = 0, 1, 2$ and 3, the bit error and block error performances of CGA(P, 2) are shown in Figure 3. For comparison, the performance of pure algebraic decoding is also included. We see that CGA(P, 2) for $P = 1, 2$ and 3 outperforms GMD algorithm (P=0) and pure algebraic decoding. At BER=10^{-4}, CGA(3,2) achieves a 0.7 dB coding gain over GMD algorithm and a 0.9 dB coding gain over pure algebraic decoding at a cost of 8 decoding trials while GMD algorithm requires 4 decoding trials. The bit and block error performances of the (255, 223) NASA standard RS code over GF(2^8) are also shown in Figure 3.

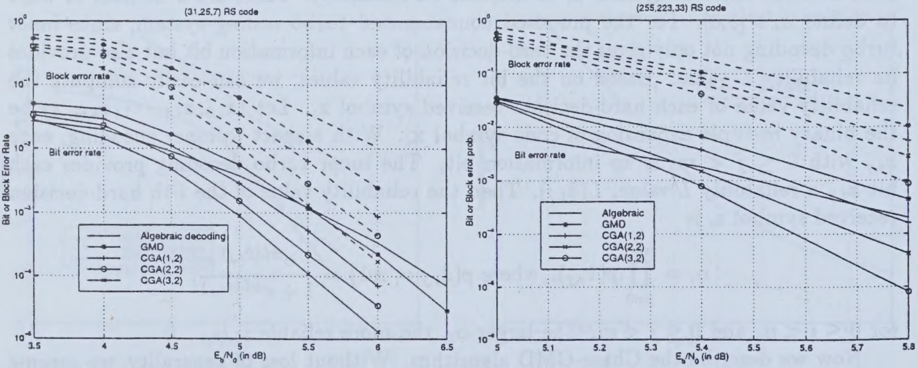


Figure 3: Error performances of Chase-GMD decoding of the (31,25,7) and the (255,223,33) RS codes

CGA(P, q') will be used in the proposed concatenated turbo coding system for decoding outer code.

4. A Concatenated Turbo Coding System

In this section, we describe the proposed concatenated turbo coding system in which the inner turbo decoding is performed in parallel mode as described in Section 2 and the outer RS code is decoded with both algebraic and CGA(P, q') algorithms. The inner code is a turbo code with two identical (n_i, k_i) block component codes C_i and the outer code is an (n_o, k_o) RS (or shortened RS) code C_o over GF(2^m) with minimum distance d. For binary transmission, each code symbol in GF(2^m) is expanded into a binary m-tuple, called an m-bit byte. Both encoding and decoding are carried out in two stages.

4.1 Encoding

An information sequence of $\lambda m k_o$ bits is segmented into a sequence of λk_o m-bit bytes. Each m-bit byte is regarded as symbol in GF(2^m). This sequence of λk_o bytes is arranged as a $\lambda \times k_o$ array $\mathbf{U}$ as shown in Figure 4(a), each column consists of λ bytes and each row consists of k_o bytes. At the first stage of encoding, each row of $\mathbf{U}$ is encoded into a RS codeword in C_o. This results in λ RS codewords arranged in a $\lambda \times n_o$ array, denoted $\mathbf{V}_1$, as shown in Figure 4(b). Let δ be the integer such that $\delta k_i = \lambda m k_o$. The interleaver between the outer encoder and the inner encoder reads the array $\mathbf{V}_1$ column by column and write row by row into $\delta \times k_i$ array, denoted $\mathbf{V}_2$, in bit form as shown in Figure 4(c). Each column consists of δ bits and each row consists of k_i bits. At the second stage of encoding, the array $\mathbf{V}_2$ is encoded into a $\delta \times (2n_i - k_i)$ array $\mathbf{V}_3$ with turbo encoding as shown in Figure 4(d) (bit demultiplexing). The turbo encoding is carried out as described in Section 2. The array $\mathbf{V}_3$ consists of three subarrays, the information array $\mathbf{V}_2$, the parity arrays $\mathbf{P}_1$ and $\mathbf{P}_2$. Each row of $\mathbf{V}_3$ is a turbo codeword which consists of k_i information bits and two parts of parity check bits, each consists of $n_i - k_i$ parity check bits. $\mathbf{V}_3$ is a concatenated turbo codeword. $\mathbf{V}_3$ is transmitted row by row. The rate of the concatenated turbo code is $R = \frac{\lambda m k_o}{\lambda m n_o + 2\lambda m n_o (n_i - k_i)/k_i}$.

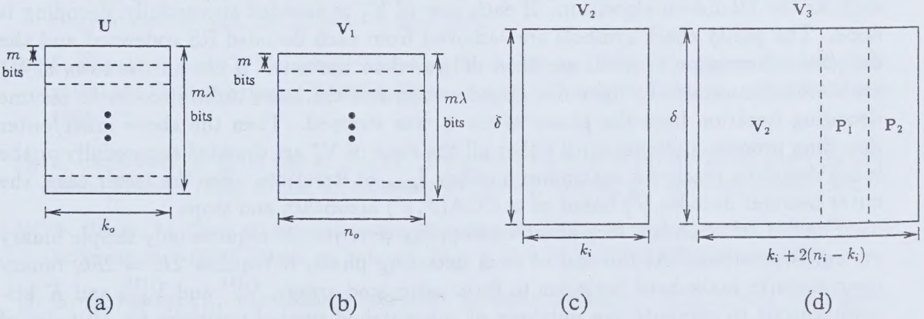


Figure 4: The structures of information and codeword arrays of the concatenated code

4.2 Decoding

Let $\mathbf{R}$ be the received array corresponding to $\mathbf{V}_3$. It is then decoded in two stages, the inner decoding and outer decoding. At inner decoding, $\mathbf{R}$ is decoded with turbo decoding in parallel mode as described in Section 2. At end of each phase of a decoding iteration, the two component decoders of the turbo decoder, DEC1 and DEC2, produce two decoded information arrays, $\hat{\mathbf{V}}_2^{(1)}$ and $\hat{\mathbf{V}}_2^{(2)}$, along with the reliability L-values of decoded bits and extrinsic values. The two estimated arrays $\hat{\mathbf{V}}_2^{(1)}$ and $\hat{\mathbf{V}}_2^{(2)}$ are compared. Using BM stopping criterion, if $\hat{\mathbf{V}}_2^{(1)}$ and $\hat{\mathbf{V}}_2^{(2)}$ match in every bit position, the turbo decoding iteration process can be stopped.

If two corresponding bits do not match at a certain bit position in $\hat{\mathbf{V}}_2^{(1)}$ and $\hat{\mathbf{V}}_2^{(2)}$, a hard-decision at this position based on $L(\hat{u}^{(1)})$ and $L(\hat{u}^{(2)})$ given in (1) is likely to result in an error. Suppose $\hat{\mathbf{V}}_2^{(1)}$ and $\hat{\mathbf{V}}_2^{(2)}$ do not match in all bit positions, we rearrange them into arrays $\hat{\mathbf{V}}_1^{(1)}$ and $\hat{\mathbf{V}}_1^{(2)}$ corresponding to the RS code array $\mathbf{V}_1$ shown in Figure 4(b). The mismatched bit positions in $\hat{\mathbf{V}}_2^{(1)}$ and $\hat{\mathbf{V}}_2^{(2)}$ will result in mismatched symbol positions in $\hat{\mathbf{V}}_1^{(1)}$ and $\hat{\mathbf{V}}_1^{(2)}$. Now we compare the corresponding rows of $\hat{\mathbf{V}}_1^{(1)}$ and $\hat{\mathbf{V}}_1^{(2)}$ and check how many symbol positions where two corresponding symbols do not match. Hard decisions at these symbol positions are likely to result in symbol errors. If the number of mismatched symbol positions for each pair of corresponding rows in $\hat{\mathbf{V}}_1^{(1)}$ and $\hat{\mathbf{V}}_1^{(2)}$ is less than or equal to the error correcting capability $t = \lfloor (d-1)/2 \rfloor$ of the outer code and if symbol errors resulting from hard decisions are only confined in these mismatched positions, then the outer RS code can be used to correct these errors. Based on this reasoning, we now can formulate a criterion for stopping the inner turbo decoding iteration and let the outer decoder to remove the remaining errors (if any).

> **Symbol Matching (SM) Stopping Criterion:** Compare row by row of $\hat{\mathbf{V}}_1^{(1)}$ and $\hat{\mathbf{V}}_1^{(2)}$. If the number of mismatched symbol positions for each pair of corresponding rows is less than or equal to $\lfloor (d-1)/2 \rfloor$, then stop the inner turbo decoding iteration.

When the inner turbo decoding is stopped based on the SM criterion, Hard decisions are made at the mismatched positions in $\hat{\mathbf{V}}_1^{(1)}$ and $\hat{\mathbf{V}}_1^{(2)}$ based on the L-values $L(\hat{\mathbf{V}}_1^{(1)})$ and $L(\hat{\mathbf{V}}_1^{(2)})$. This results in an estimated array $\mathbf{V}_1^*$. Then second decoding stage starts. The outer decoder decodes each row of $\mathbf{V}_1^*$ based on an algebraic decoding algorithm, such as the Euclidean algorithm. If each row of $\mathbf{V}_1^*$ is decoded successfully, decoding is done. The parity check symbols are removed from each decoded RS codeword and the decoded information symbols are then delivered to the user. If not all the rows of $\mathbf{V}_1^*$ are decoded successfully, the outer decoder instructs the inner turbo decoder to resume decoding iteration from the phase where it was stopped. Then the above inner/outer decoding process continues until either all the rows of $\mathbf{V}_1^*$ are decoded successfully or the inner decoding reaches a maximum number $I_{\max}$ of iterations. For the latter case, the outer decoder decodes $\mathbf{V}_1^*$ based on a CGA(P, q') algorithm and stops.

The SM criterion is a very effective stopping criterion. It requires only simple binary or logic operations. At the end of each decoding phase, it requires $2K = 2\delta k_i$ binary operations to make hard decisions to form estimated arrays, $\hat{\mathbf{V}}_1^{(1)}$ and $\hat{\mathbf{V}}_1^{(2)}$, and K bit-comparisons to compute the numbers of mismatched symbol positions for all pairs of corresponding rows in $\hat{\mathbf{V}}_1^{(1)}$ and $\hat{\mathbf{V}}_1^{(2)}$.

5. An Example and Simulation Results

The proposed interactive concatenated turbo coding system has been simulated for both AWGN and Rayleigh channels. Simulation results show that this coding system achieves both good bit-error and frame error performances without error floor. Furthermore, the SM stopping criterion effectively terminates the inner turbo decoding iteration and shortens the inner decoding delay.

Consider an example in which the (228, 212) shortened RS code over GF(2^8) is used as the outer code and the (64,57) distance-4 extended Hamming code is used as the two component codes for constructing the inner turbo code. We choose $\lambda = 4$. Then $\delta = 128$. The rate of this system is R=0.75. The bit-error and frame-error performances of this system for AWGN channel are shown in Figure 5. We see the waterfall performance without error floor. For the Rayleigh channel without channel side information, the bit-error and frame-error performances are shown in Figure 6. Again they display waterfall error performance. Form Figure 5, we see that for AWGN channel and at BER=10^{-4}, the proposed iterative decoding achieves a 5.5 dB coding gain over the uncoded BPSK system which is only 1.2 dB away from the Shannon limit for rate $R = 0.75$. To show the effectiveness of the SM criterion, Figure 7 displays the average numbers of decoding iterations required with SM and CE criteria, respectively, for the example system over AWGN channel. From Figures 2, 5 and 7, we see that SM criterion is more effective than the CE criterion. It reduces the number of decoding iterations with very little performance degradation.

6. Conclusion

In this paper, we have presented a high performance concatenated turbo coding system in which the inner and outer decoders interact to achieve good error performance. An effective criterion for stopping inner turbo decoding iteration has been proposed, which is more effective than the CE criterion in reducing the number of decoding iterations. Also presented in this paper is a reliability-based decoding algorithm which provides a good trade-off between the error performance of Chase-2 algorithm and complexity of GMD decoding algorithm.

References

[1] C. Berrou, A. Glavieux and P. Thitimajshima, " Near Shannon Limit Error-Correcting Coding and Decoding: Turbo Codes," *Proc. of IEEE Intl. Conf. on Communications*, Geneva, Switzerland, pp. 1064-1070 May 1993.

[2] S. Benedetto and G. Montorsi, "Unveiling Turbo Codes: Some Results on Parallel Concatenated Coding Schemes," *IEEE Trans. Inform. Theory*, Vol. 42, pp. 409-428. March 1996.

[3] J. Hagenauer, E. Offer and L. Papke, "Iterative Decoding of Binary Block and Convolutional Codes," *IEEE Trans. on Information Theory*, Vol. 42, pp. 429-445, March 1996.

[4] D. Chase, "A Class of Algorithms for Decoding Block Codes with Channel Measurement Information," *IEEE Trans. on Inform. Theory*, Vol. 18, pp. 170-182, January 1972.

[5] G. D. Forney, Jr., "Generalized Minimum Distance Decoding," *IEEE Trans. on Inform. Theory*, Vol. 12, pp. 125-131, April 1966.

[6] D. Divsalar and F. Pollara, "Turbo Codes for PCS Applications," *Proc. of ICC'95*, Seattle, WA, June 1995.

[7] Y. Shao, M. P. C. Fossorier and S. Lin, "Two Simple Stopping Criteria for Iterative Decoding," *IEEE Trans. on Communications*, to appear 1999.

[8] G. Einarsson and C. Sundberg, "A Note on Soft Decision Decoding with Successive Erasures," *IEEE Trans. on Information Theory*, Vol 22, pp. 88-96, January 1976.

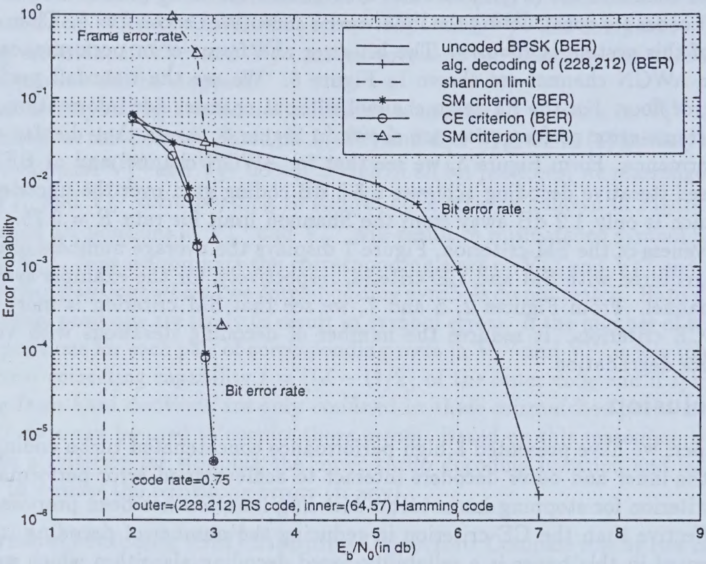


Figure 5: Bit and frame error performances of the example system over the AWGN channel

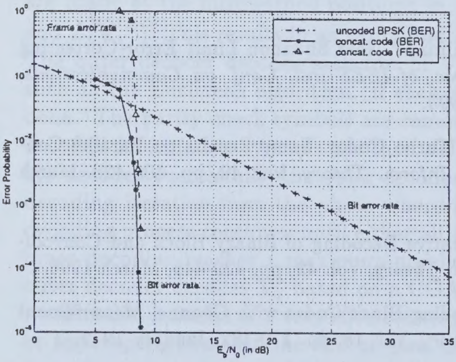


Figure 6: Bit and frame error performances of the example system over Rayleigh fading channel

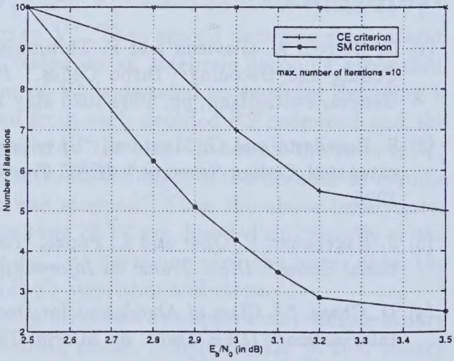


Figure 7: Number of iterations of MSN and CE criteria of the example system over AWGN channel

Bi-directional SOVA Decoding for Turbo-codes

Jinghu Chen, Marc P. C. Fossorier, Shu Lin
Dept. of Electrical Engineering
University of Hawaii
2540 Dole Street
Honolulu, HI 96822
USA
jinghu, marc, slin@spectra.eng.hawaii.edu

Chengqi Xu
Dept. of Information Engineering
Nanjing University of Posts & Telecommunications
Nanjing 210003
China
xucq@em.njupt.edu.cn

Abstract

In this paper, we present a bi-directional SOVA-based decoding scheme for turbo-codes, which combines the soft-outputs provided by both forward and backward SOVA decodings. The complexity of this bi-directional SOVA decoding only doubles that of the conventional SOVA, but can achieve about the same performance as the Max-Log-MAP algorithm in turbo decoding with much less computational complexity.

1 Introduction

The symbol-by-symbol *maximum a posteriori* (MAP) algorithm, introduced in [1], can be adopted in a straightforward way for turbo decoding and achieve very good performance [2][3][4]. Unfortunately, the complexity of the MAP algorithm is very high. The Soft Output Viterbi Algorithm (SOVA), proposed in [5], can be implemented for turbo decoding with moderate complexity [4][6]. However, the coding gain achieved by the SOVA is generally about 0.7 dB less than that of MAP decoding [7]. The MAP algorithm can also be approximated by the Max-Log-MAP algorithm [4][7], which avoids logarithmic and exponential operations at the expense of a few tenths of a dB degradation in error performance. Both Max-Log-MAP and SOVA can be viewed as sub-optimal reduced complexity versions of the MAP algorithm.

Some research has been done to improve the performance of SOVA for turbo decoding [8]-[11]. In this paper, we propose a decoding scheme based on the original SOVA of [5]. This decoding scheme employs an independent backward SOVA decoding in addition to the forward one. The hardware complexity only doubles that of the original one, with very limited additional delay for turbo decoding. Simulations show that bi-directional SOVA not only significantly outperforms the original SOVA for turbo decoding, but surprisingly can also occasionally slightly outperforms the iterative Max-Log-MAP algorithm. Most importantly, the reasons of the close performances achieved by bi-directional SOVA and Max-Log-MAP decodings are elaborated.

2 A Brief Review of MAP, Max-Log-MAP and SOVA Algorithms

From the viewpoint of trellis representation, for a given time index i, the MAP algorithm divides the set of paths into two subsets, $\Omega_1(i)$ and $\Omega_0(i)$, which consist of all the paths corresponding to the information bit $u_i = 1$ and $u_i = 0$, respectively. Then it calculates the sum of the probabilities of all the paths in each subset, conditioned on the received vector $\mathbf{r}$, and finally computes the logarithmic likelihood ratio Λ_i of u_i, *i.e.*,

$$\Lambda_i = \ln\left(\sum_{\mathbf{c}\in\Omega_1(i)} P(\mathbf{c}\mid\mathbf{r})\right) - \ln\left(\sum_{\mathbf{c}\in\Omega_0(i)} P(\mathbf{c}\mid\mathbf{r})\right) \tag{1}$$

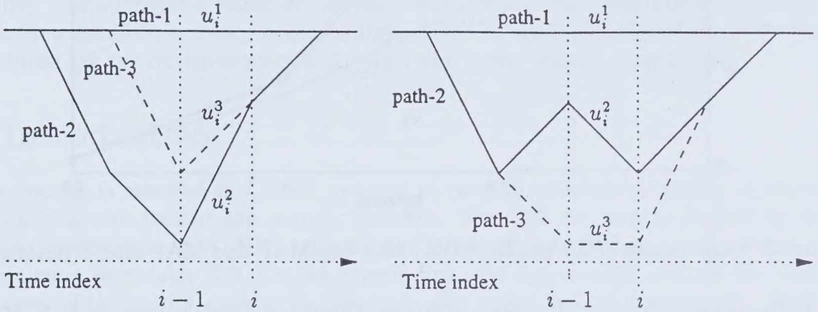
For each i, both Max-Log-MAP and SOVA decodings consider only two paths to evaluate the soft outputs, one from $\Omega_1(i)$ and the other from $\Omega_0(i)$. The Max-Log-MAP algorithm always chooses one path from $\Omega_1(i)$ with the largest probability conditioned on $\mathbf{r}$ (we call it the best path in $\Omega_1(i)$), and one path from $\Omega_0(i)$ with the largest probability conditioned on $\mathbf{r}$ (we call it the best path in $\Omega_0(i)$). It is readily seen that one of the two best paths must be the maximum likelihood (ML) decoding path. Unfortunately, the SOVA only guarantees to find one best path (the ML path). The other path is not necessarily the best one, due to the fact that the best path with opposite u_i to the ML path may be discarded before it remerges with the ML path. For example, in Figure 1-(a), suppose: (i) path-1 is the ML path with information bit $u_i^1 = 0$; (ii) path-3 is the best path with $u_i^3 = 1$; and (iii) path-2 survives at time-i and remerges with path-1 later. In this case, the SOVA does not consider path-3, as it has been discarded at time-i. It follows that in general, the soft outputs delivered by the SOVA are over-estimated with respect to those of the Max-Log-MAP decoding. These inaccurate reliability values degrade the performance of the iterative turbo decoding.

3 Bi-directional SOVA for Turbo-decoding

Given a received sequence $\mathbf{r}$, both the VA and the SOVA can start decoding simultaneously the first received value and the last received value, from both ends of the terminated trellis. The hard decisions of the forward and backward decodings are the same as in both cases, the same ML path is obviously chosen. We also find in simulations that if we replace all the forward SOVA decoders in turbo decoding by the backward ones, there is no degradation nor improvement in performance, as expected. As a result, the backward SOVA has no advantages over the forward one if they are processed independently. However, there are differences between the soft output values delivered by the forward and backward SOVA's, which suggests to exploit these differences to improve the soft outputs of the SOVA.

The differences in the soft outputs between forward and backward SOVA's result from the different selections of paths in calculating the reliability $|\Delta|$. As stated in Section 2, for a given i, the original forward SOVA may not choose one of the two best paths, either in $\Omega_1(i)$ or $\Omega_0(i)$, because this path is discarded before it remerges with the ML path. However, the best path discarded by the forward SOVA can survive until it remerges with the ML path in the backward SOVA, and the resultant Δ provides a more reliable value about u_i. Unfortunately, there are also cases in which the SOVA will fail to choose the best path in both directions. These two possible cases are illustrated in Figures 1-(a) and

1-(b), respectively. In Figure 1-(a), path-3 discarded by the forward SOVA at time-i can survive until it remerges with the ML path (*i.e.* path-1) in the backward SOVA. However, in Figure 1-(b), path-3 is discarded in both ways. Also even if the best path does not survive in both SOVA decodings, the backward SOVA may still find another path better than that in forward SOVA, and get a more accurate soft-output for u_i. If the reliability value of u_i in backward SOVA is less than that obtained in forward SOVA, we simply substitute the former for the later to improve the soft-output of the SOVA. This decoding is referred to as bi-directional SOVA (bi-SOVA), and can be applied to turbo decoding. Since the forward and backward SOVA's can be carried out parallel, the total extra delay is only that spent on comparing and selecting the soft-values, which is obviously very small.



(a) (b)

Figure 1: Possible cases of path-selecting in decoding with SOVA.

4 Simulation Results

Simulations have been conducted to compare the performance of bi-SOVA with those of MAP, SOVA and Max-Log-MAP in turbo decoding. Figure 2 depicts the BER's with these four decodings after 8 iterations. The conditions of the simulations are the following. We used two component codes, each with rate $R = 1/2$ and generator polynomial $G = (37,\ 21)$. The interleaver size is $N = 1024$, with the so-called non-uniform interleaver introduced in [2]. All algorithms assume no finite-length decoding window.

References

[1] L. R. Bahl and J. Cocke, F. Jelinek, and J. Raviv, "Optimal Decoding of Linear Codes for Minimizing Symbol Error Rate", *IEEE Trans. Inform. Theory*, vol. 20, pp. 248-287, Mar. 1974.

[2] C. Berrou and A. Glavieux, "Near Optimum Error-Correcting Coding and Decoding: Turbo-codes", *IEEE Trans. Commun.*, vol. 44, pp. 1261-1271, Oct. 1996.

[3] P. Robertson, "Illuminating the Structure of Code and Decoder of Parallel Concatenated Recursive Systematic (Turbo) Codes", *Proc. GLOBECOM94*, pp. 1298-1303, Dec. 1994.

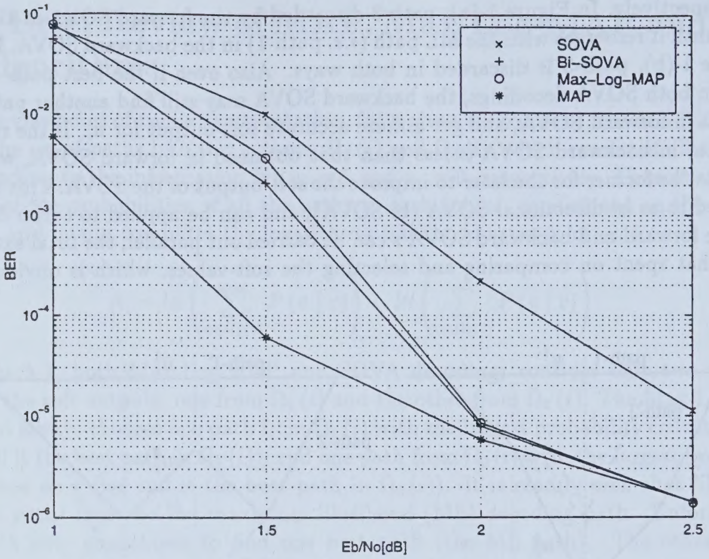


Figure 2: Performance of SOVA, Bi-SOVA, Max-Log-MAP and MAP after 8 iterations.

[4] J. Hagenauer, "Iterative Decoding of Binary Block and Convolutional Codes", *IEEE Trans. Inform. Theory*, vol. 42, pp. 429-445, Mar. 1996.

[5] J. Hagenauer and P. Hoeher, "A Viterbi Algorithm with Soft-Decision Outputs and its Applications", *Proc. GLOBECOM89*, pp. 1680-1686, Nov. 1989.

[6] J. Hagenauer, "Source-Controlled Channel Decoding", *IEEE Trans. Commun.*, vol. 43, pp. 2449-2457, Sept. 1995.

[7] P. Robertson, E. Villebrun and P. Hoeher, "A Comparison of Optimal and Sub-Optimal MAP Decoding Algorithms Operating in the Log Domain", *Proc. ICC95*, pp. 1009-1013, June 1995.

[8] G. Battail, "Ponderation des Symboles Decodes par l' Algorithme de Viterbi", *Ann. Telecommun.*, vol. 42, pp. 31-38, Jan. 1987

[9] L. Lin and R. S. Cheng, "Improvements in SOVA-based decoding for turbo codes", *Proc. ICC97*, pp. 1473-1478, June 1997.

[10] M. P. C. Fossorier, F. Burkert, S. Lin, and J. Hagenauer, "On the Equivalence Between SOVA and Max-Log-MAP Decodings", *IEEE Commun. Letters*, Vol. 2, pp. 137-139, May 1998.

[11] L. Papke and P. Robertson, "Improved Decoding with the SOVA in a Parallel Concatenated (Turbo-code) Scheme", *Proc. ICC96*, pp. 102-106.

Analysis of an Up/down Power Control Algorithm in CDMA Reverse Link Under Fading Condition *

Lei Song† Narayan Mandayam† and Zoran Gajic
†Wireless Information Network Laboratory (WINLAB)
Rutgers University, 73 Brett Road, Piscataway, NJ 08854

Abstract In this paper, we propose an analytical approach to study the performance of a CDMA reverse link with an up/down power control algorithm in the presence of fading. The approach is based on the technique of statistical linearization. Numerical results show excellent accuracy of our analysis. Such an analytical approach can be used to optimize the system parameters and design new power control algorithms.

1 Introduction

Power control is essential in CDMA systems to provide satisfactory quality of service (QoS) and to also combat the near-far problem. The QoS for voice is decided by the signal to interference ratio (SIR) achieved and is usually specified as a requirement on the minimum acceptable SIR. On the reverse link, the base station controls the transmit power of the mobile stations so that each user meets its QoS requirement. Unlike the forward link, it is not possible to obtain orthogonal spreading. Ideally, control is accomplished based on the knowledge of the SIR achieved by the mobile. Power control schemes can be centralized [1] or decentralized [2] according to the nature of control. A centralized controller has the information (e.g., all link gains) for each user and it decides control actions for all the users [1]. A decentralized controller on the other hand, only utilizes local information to decide the control action for a user (e.g., only uses a user's own link gain and/or SIR to decide its power) [2].

The study of the performance of the above algorithms (in the literature) often assumes the availability of link information (such as channel gains or SIR values) and its exact exchange between the base station and mobile station. However, in practical systems such as [3], the link information is not readily available and the exchange between the base station and mobile station is limited to qualitative information depending on the SIR level achieved. As a result, power control is usually done based on (feedback) commands from the base station where the transmit power of the mobile is either increased or decreased. We will refer to this kind of power control as *up/down* power control. In this type of power control, the base station measures the SIR for each user and compares it with a target. A power control bit is then sent to the mobile station on the feedback channel once in every predetermined epoch referred to as the power control group (PCG) [3].

This paper analyzes the reverse link of a CDMA system with this up/down power control algorithm. Stability and convergence properties of a similar power control loop

---

*This work is supported in part by NSF under Grant no. NCR 97-06036

has been studied in [4] for nonfading environments. In this paper, we will study the performance of the up/down algorithm in a fading environment. We will first model the power controlled reverse link as a stochastic nonlinear feedback control system. Then the performance of the above power control algorithm is studied using the technique of statistical linearization. Numerical results show excellent accuracy of the approximation. Such an analytical approach can be used to design/optimize the system parameters without going through lengthy simulations. For example, our analysis gives a direct method to evaluate power control stability and errors as a function of mobile speed, power control step size and fading channel parameters such as correlation coefficient and rate of fading.

2 System Model

We consider a CDMA uplink with K users. The notation we use here resembles that in [5]. In general the received signal is given by

$$r(t) = \sum_{k=1}^{K} \sum_{j=-\infty}^{\infty} \sqrt{2p_k(t)} b_{k,j} c_{k,j} s_k(t - jT_c - \tau_k) e^{j\phi_k} + N(t) \tag{1}$$

T_c is the chip duration. $c_{k,j}$ is the value of the jth chip of the k^{th} user. $b_{k,j}$ represents the value of the bit containing the jth chip $c_{k,j}$. It takes on the same value for M successive chips where M is the number of chips in a bit. $s_k(t)$ is the chip waveform. ϕ_k is the phase offset and τ_k is the delay. $p_k(t)$ is the received power of user k which is equal to $a_k(t)x_k(t)$. $x_k(t)$ is the mobile transmit power and $a_k(t)$ is the link gain. Since both the mobile transmit power and the link gain are assumed to be fixed during a Power Control Group (PCG), (which contains multiple chips), the time variable t can be ignored in our analysis. $N(t)$ is a zero mean Gaussian process with two-sided power spectral density $N_0/2$.

We are interested in user 1, whose delay is assumed to be 0. For other users, τ_k is *i.i.d* uniform over $[0, T_c]$. We will consider a BPSK system with perfect phase estimation. The bits and chips are +1 and -1 with equal probability. The receiver is assumed to be a matched filter matched to the spreading waveform $c_1(t)$. The corresponding detection statistic is

$$D_1 = \sqrt{2p_1} M b_1 + \sum_{k=2}^{K} \sum_{m=0}^{M-1} c_{1,m} r_{k,m} + N_k \tag{2}$$

where $r_{k,m} = \sum_{j=-\infty}^{\infty} \sqrt{2p_k} c_{k,j} R_s(j+\epsilon_k-m)$ and $N_k \sim N(0, M\frac{N_0}{2})$. Let $I_k = \sum_{m=0}^{M-1} c_{1,m} r_{k,m}$,

$$E\{I_k^2\} = 2Mp_k \sum_{j=-\infty}^{\infty} \int_{-\infty}^{\infty} R_s^2(j + \epsilon_k) d\epsilon_k = 2Mp_k\sigma_s^2, \tag{3}$$

where $R_s(\xi)$ is the correlation of waveforms separated by ξT_c and $\sigma_s^2 = \int_{-\infty}^{\infty} |\Psi(f)|^4 df$. $\Psi(f)$ is the spectrum of the chip pulse waveform. For a rectangular pulse which has a *sinc* spectrum, $\sigma_s = 2/3$ [6, 7]. For a *sinc* pulse with rectangular spectrum, $\sigma_s = 1$ [5]. A proof of this result can be found in Theorem 1 of [5]. In this paper, we will consider a system with a relaxed bandwidth constraint. Finally, we have

$$E\{I_k^2\} = \frac{4}{3} M p_k \tag{4}$$

Instead of considering the above quantities as functions of time t, we can index them by the PCGs. Then the SIR for user 1 can be written in discrete form as

$$\gamma_1(n) = \frac{2p_1(n)M}{\frac{4}{3}\sum_{k=2}^{K} p_k(n) + N_0/2} = \frac{2a_1(n)x_1(n)M}{\frac{4}{3}\sum_{k=2}^{K} p_k(n) + N_0/2} \tag{5}$$

To facilitate a tractable analysis, we will assume that $p_k(n)$ is controlled around the target p_k^* and replace $\sum_{k=2}^{K} p_k(n)$ by $\sum_{k=2}^{K} p_k^*$. In [4], we showed that $p_k(n)$ will converge to a neighborhood of p_k^* for feasible systems with constant link gains. Essentially, we assume that the sum of a large number of interferers is relatively constant during a frame. Note that we do not need to estimate p_k^*. We will show that our proposed analytical results are valid as long as the sum remains close to a constant. Such treatment of $p_k(n)$ is also consistent with the results in [8] as well as experimental measurements of [9, 10]. This approximation removes the correlation in the received power from interferers and simplifies the problem. Finally, equation (5) is reduced to

$$\gamma_1(n) = \frac{2a_1(n)x_1(n)M}{\frac{4}{3}\sum_{k=2}^{K} p_k^* + N_0/2} \tag{6}$$

We will use the capital letters to denote values in dB. In dB, equation (6) can be written as

$$\Gamma_1(n) = A_1(n) + X_1(n) + I \tag{7}$$

where I is a constant given by

$$I = 10\log\frac{2M}{\frac{4}{3}\sum_{k=2}^{K} p_k^* + N_0/2} \tag{8}$$

Finally, we can write the state equation for the up/down power control algorithm as a nonlinear difference equation

$$X(n+1) = X(n) + d\Psi(\Gamma^{tar} + I - X(n) - A(n)) \tag{9}$$

where the nonlinear term Ψ is defined as

$$\Psi(\xi) = \begin{cases} 1 & \text{if } \xi \geq 0 \\ -1 & \text{if otherwise} \end{cases} \tag{10}$$

Since user 1 is the user being considered, we drop the subscript 1 for simplicity. The block diagram is shown in Figure 1. The input to the nonlinear system is the link gain sequence

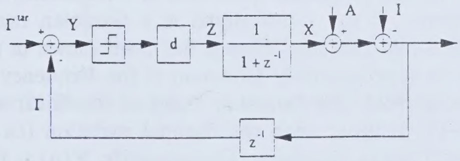


Figure 1: Block diagram for the up/down power control algorithm

$\{A(n)\}$. We assume shadowing as the main factor in deciding the characteristics of the channel gain sequence used as the input to system (9). A simple and realistic model of $A(n)$ is a Gaussian process with the correlation given by Stüber in [11]

$$R_A(n) = \sigma_A^2 \xi_D^{(vT/D)|n|} \tag{11}$$

ξ_D is the correlation for a spatial distance D. T is the sampling period and v is the mobile speed. Also note that in (9), we can combine A with I and Γ by defining $B(n) := A(n) - I - \Gamma^{tar}$. $B(n)$ is still a Gaussian process with the same covariance property as $A(n)$.

3 Analysis: Statistical linearization

To evaluate the performance of the power control algorithm, we are interested in finding the distribution of the SIR error $Y(n) = \Gamma(n) - \Gamma^{tar}$. A narrow *pdf* centered at 0 implies tight power control. However, to find the *pdf* of $Y(n)$ from Equation (9) is not simple. The *pdf* can be found using the method of Markov Process outlined in ([12], Chapter 1). This method is the utmost method in providing an analytical nonlocal solution for a general nonlinear system. However, it needs to solve the Fokker-Planck-Kolmogorov (PFK) equation, which is quite difficult. Analytical solutions are possible only for some limited cases. As an alternative, local methods such as perturbation method and statistical linearization may be used [12, 13]. Perturbation method is not feasible for this power control problem due to the discontinuity of $\Psi(\,)$ at 0. In this paper, we will analyze the system mainly through statistical linearization.

The statistical linearization approach [14] is based on the concept of approximating the solution to a nonlinear random system by that obtained from an "equivalent" linear system. The equivalent linear system is obtained by replacing the nonlinear component with a linear one which minimizes a certain objective function such as mean square error within a certain class of linear approximators. The best linear approximation depends on the statistics of the input signal to the nonlinear component. On the other hand, the statistics of the input signal can also be found by solving the equivalent linear system once the best linear approximator is decided. Thus, we can set up an equation for the input signal statistics. An inherent difficulty of this approach is that this equation not only contains the first and second moments but higher moments as well and some time even the entire pdf is needed. To overcome this difficulty, some hypothesis is usually introduced concerning the nonlinear input signal pdf. And the functional equation is simplified into an algebraic or transcendental equation of some parameters such as low order moments.

Let us consider a feedback system consisting of a nonlinear memoryless component and a linear component. If the input signal is a Gaussian random process and the nonlinearity is a lagless odd function, then it has been shown in ([13], section 4.6) that the system response is approximately Gaussian if the frequency band of the external signal does not coincide with the frequency band of the linear component. Since the power adaptation in (9) is faster than the channel variation (i.e., the external signal), we shall approximate $X(n)$ as Gaussian. Consequently, $Y(n)$ is Gaussian as well. This assumption is also consistent with the classical treatment in [8]. We can further argue that $Y(n)$ has zero mean.

The simplest linear approximation is to replace a nonlinear component $f(x)$ with a constant gain. The best approximator K^* minimizes an objective function such as

$E\{(Kx - f(x))^2\}$. This simple approximation was shown to generate reasonably good results for continuous nonlinear systems in [12, 13]. However, we find that for the discrete nonlinear system such as (9), this simple linearization drastically underestimates the variance of $Y(n)$. Instead, we propose to approximate $d\Psi(\ )$ with a constant gain K and an uncorrelated random noise with the restriction that the equivalent term results in the same the mean and variance as the original nonlinear term. In other words, we want to approximate the output of $Z = d\Psi(Y)$ with $\hat{Z} = KY + W$ for some constant gain K and white noise W with a constraint that Z and $\hat{Z}$ have the same mean and variance. This approximation is shown in Figure 2. Within this class of linear approximators, the optimal one is defined as the one that minimizes the MSE $E\{(Z - KY)^2\}$. Note that Z being zero mean implies that W has zero mean as well.

Let $Y \sim N(0, \sigma_Y^2)$. Assuming σ_Y is known, we shall find the optimal K and σ_W as functions of σ_Y. To find K^*, we proceed as follows:

$$\begin{aligned} E\{(Z - KY)^2\} &= E\{Z^2\} + K^2E\{Y^2\} - 2KE\{YZ\} \\ &= 1 + K^2\sigma_Y^2 - 2KE\{YZ\} \end{aligned} \tag{12}$$

Taking the partial derivative,

$$\frac{\partial E\{(Z - KY)^2\}}{\partial K}|_{K=K^*} = 2K^*\sigma_Y^2 - 2E\{YZ\} = 0 \tag{13}$$

This yields

$$K^* = \frac{E\{YZ\}}{\sigma_Y^2} = \frac{\sqrt{\frac{2}{\pi}}d}{\sigma_Y} \tag{14}$$

The variance of W can be found using the constraint that Z and $\hat{Z}$ have the same variance.

$$E\{\hat{Z}^2\} = (K^*)^2E\{Y^2\} + E\{W^2\} = E\{Z^2\}, \tag{15}$$

which implies

$$E\{W^2\} = (1 - \frac{2}{\pi})d^2 \tag{16}$$

Replacing $d\Psi(Y)$ with $K^*Y + W$, we have the equivalent linear system with the block diagram shown in Figure 2. We use two approaches to find σ_Y in terms of K^*, namely

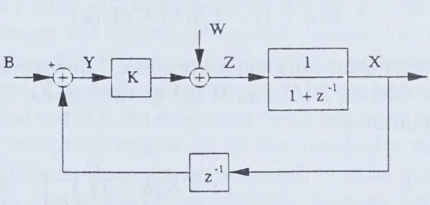


Figure 2: Block diagram for the equivalent linear system

(1) The spectrum integration approach and (2) The Lyapunov equation approach.

3.1 Spectrum Integration Approach

The transfer function from B to X is

$$H_{B,X}(e^{-jw}) = \frac{e^{-jw}}{1-(1-K^*)e^{-jw}} \tag{17}$$

and thus the transfer function between B and Y is

$$H_{B,Y}(e^{-jw}) = 1 - H_{B,X}(e^{-jw}) = \frac{1-e^{-jw}}{1-(1-K^*)e^{-jw}} \tag{18}$$

The transfer function from W to Y is

$$H_{W,Y} = \frac{-1}{1-K^*e^{-jw}} \tag{19}$$

Thus the variance of Y is given as

$$E\{Y^2\} = \frac{1}{2\pi}\int_{-\pi}^{\pi} |H_{B,Y}(e^{-jw})|^2 S_B(e^{-jw})dw + \sigma_W^2 \int_{-\pi}^{\pi} |H_{W,Y}(e^{-jw})|^2 dw \tag{20}$$

where $S_B(e^{-jw})$ is the spectrum of B and is given by

$$S_B(e^{-jw}) = \frac{1}{1+\xi_D^{2vT/D}} \tag{21}$$

The variance of Y can be solved numerically once the spectrum of B is known. We can further argue that the solution always exists.

3.2 Lyapunov Equation Approach

If the fading can be further modeled as a Gaussian white noise process filtered by a low pass filter, the power control error variance can be obtained more efficiently using the Lyapunov equation [15]. Note that the spectrum of $B(k)$ is exponential. A more general model for a random process with exponential spectrum is

$$B(k+1) = \xi B(k) + \eta U(k).$$

where $U(k)$ is a zero-mean Gaussian random process. The variance of power control error Y can be found by considering $U(k)$ and $W(k)$ as two inputs.

The state space equation is

$$\begin{bmatrix} X(k) \\ X(k+1) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\xi(1-K) & 1-K+\xi \end{bmatrix} \begin{bmatrix} X(k-1) \\ X(k) \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 \\ -K\eta & K\xi & K \end{bmatrix} \begin{bmatrix} U(k) \\ W(k-1) \\ W(k) \end{bmatrix}$$

and the output equation is

$$Y(k) = [\frac{1}{K} \ -\frac{1}{K}] \begin{bmatrix} X(k) \\ X(k+1) \end{bmatrix} + [0 \ 0 \ -K] \begin{bmatrix} U(k) \\ W(k-1) \\ W(k) \end{bmatrix} \tag{22}$$

Let V_X be the covariance of $\underline{X}(k) = \begin{bmatrix} X(k-1) \\ X(k) \end{bmatrix}$, then $V_X = \begin{bmatrix} \sigma_X^2 & R_X(1) \\ R_X(1) & \sigma_X^2 \end{bmatrix}$. Let V_I be the covariance of the input $\begin{bmatrix} U(k) \\ W(k-1) \\ W(k) \end{bmatrix}$, then $V_I = \begin{bmatrix} \sigma_U^2 & 0 & 0 \\ 0 & \sigma_W^2 & 0 \\ 0 & 0 & \sigma_W^2 \end{bmatrix}$. For simplicity, let us denote $a = -\xi(1-K)$ and $b = 1-K+\xi$. Let $H = \begin{bmatrix} 0 & 1 \\ a & b \end{bmatrix}$ and $G = \begin{bmatrix} 0 & 0 & 0 \\ -K\eta & K\xi & K \end{bmatrix}$. The covariance matrix of $\underline{X}$ satisfies the following equation.

$$V_X = HV_XH^T + GV_IG^T \tag{23}$$

This is an algebraic Lyapunov equation [15]. The solution is given as

$$\sigma_X^2 = \frac{(1-a)(K^2\eta^2\sigma_U^2 + K^2\xi^2\sigma_W^2 + K^2\sigma_W^2)}{1-a^2-b^2-ab^2+a^3} \tag{24}$$

and

$$R_X(1) = \frac{b}{1-a}\sigma_X^2 \tag{25}$$

The variance of Y can be found as

$$\begin{aligned} \sigma_Y^2 &= \frac{2}{K^2}(\sigma_X^2 - R_X(1)) + \sigma_W^2 \\ &= \frac{\sigma_W^2}{1-(1-K)^2} + \frac{2(1-a-b)(1-\xi)^2\sigma_U^2}{1-a^2-b^2-ab^2+a^3} \end{aligned} \tag{26}$$

The above equation can be rewritten as a fifth order polynomial equation for σ_Y. Then σ_Y can be solved using the fixed point iteration method.

4 Numerical Results

We first analyze how the up/down algorithm performs for different speeds. The power control error standard deviation σ_Y v.s speed v is plotted in Figure 3. σ_Y vs channel correlation ξ is shown in Figure 4. The solid curves are the simulation results for different number of users. In all cases, the systems are equally loaded, i.e., the sum of SIR targets is fixed. The target total received power is 9 dB over the noise floor σ^2 and all the users have the same SIR target. The dash-dotted curve is the analytical result using spectrum integration and the dashed curve is the analytical result obtained by solving the Lyapunov equations. Overall, the analytical results match the simulation results results very well. This shows that our statistical linearization approach is valid and accurate. The 1-user simulation results match that of the 16-user system, there by justifying the assumption of constant sum of interferers. The second test is to see how the power control algorithm works with different step size d. Assuming the same parameters for the channel and a mobile speed of 60 kmph, σ_Y is plotted in Figure 5. In simulation, we assume there are 16 users in the system. Again, it shows good match. An interesting question is to find the optimal step size d for users with different speeds. Simulation is very time consuming. The analytical approach using Lyapunov equation is very efficient. The optimal step size for different speeds are plotted in Figure 6. An important observation is that, the optimal step size depends on the mobile speed, or the channel correlation factor. We can

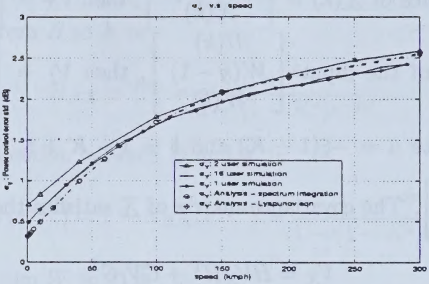


Figure 3: σ_Y v.s v ($d = 0.5$ dB, $D = 1$ m, $\xi = 0.1$, $T = 1.25$ ms. $\sigma_B = 3$ dB.)

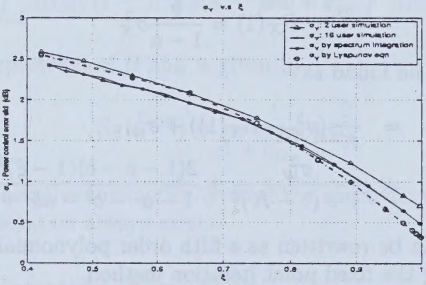


Figure 4: σ_Y v.s. correlation ξ ($d = 0.5$ dB, $D = 1$ m, $\xi = 0.1$, $T = 1.25$ ms. $\sigma_B = 3$ dB.)

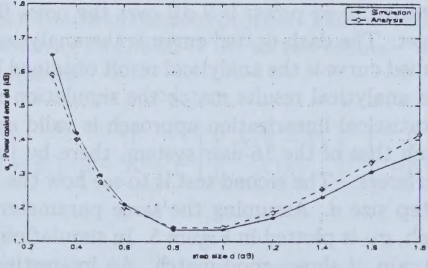


Figure 5: σ_Y v.s. d ($v = 60$ kmph, $D = 1$ m, $\xi = 0.1$, $T = 1.25$ ms. $\sigma_B = 3$ dB.)

see that at low speed, a smaller step size should be used since the fading is slow. It can be shown using the Lyapunov method that if the speed is too high, the optimal step size decreases again (not shown here). It is meaningful to know if the statistical linearization technique can be applied to other fading channels such as those suggested by ITU in [16]. As an example, we consider a one-path Rayleigh fading channel. Statistical linearization is used to obtain the analytical results. In Figure 7, σ_Y by analysis is compared with that obtained by simulation for a 16-user system. Both analysis and simulation show that the up/down power control is effective only when then mobile speed is slow.

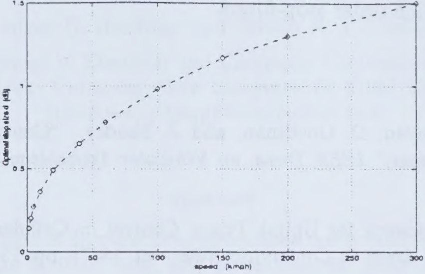


Figure 6: Optimal step size d v.s. speed

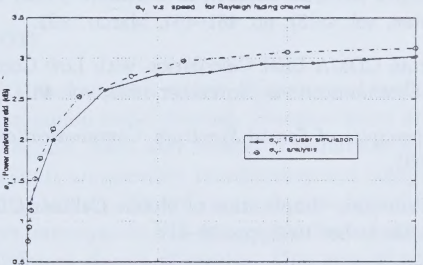


Figure 7: σ_Y v.s speed for one-path Rayleigh channel

5 Conclusion

In this paper, a CDMA reverse link with the up/down power control algorithm was studied in the presence of fading. The effect of mobile speed, power control step size and fading channel parameters such as correlation coefficient and rate of fading on power control errors was investigated. First, we derived a stochastic nonlinear feedback control system model for the power controlled reverse link. And then, we studied the power control performance based on the nonlinear model using the technique of statistical linearization. Numerical results show excellent accuracy of our analysis. Such an analytical approach can be used to design and optimize the system parameters without going through lengthy simulations. For example, our analysis gives a direct method to evaluate power control stability and errors as a function of mobile speed, power control step size and fading channel parameters such as correlation coefficient and rate of fading. Such an approach is also useful in designing more advanced power control schemes such as in systems with multi-bit feedback or high order loop-filters.

References

[1] S. Grandhi, R. Vijayan, D. Goodman, and J. Zander, "Centralized Power Control in Cellular Radio Systems," *IEEE Trans. on Vehicular Technology*, vol. 42(4), pp. 466 – 468, 1993.

[2] R. Yates, "A Framework for Uplink Power Control in Cellular Radio System," *IEEE Journal on Selected Areas in Communications*, vol. 13(7), pp. 1341–1347, 1998.

[3] TIA/EIA/IS-95, *Mobile Station-Base Station Compatibility for Dual-Mode Wideband Spread Spectrum Cellular System*, Telecommunication Industry Association, 1993.

[4] L. Song, N. Mandayam, and Z. Gajic, "Analysis of an Up/Down Power Control Algorithm for CDMA Reverse Link: A Nonlinear Control System Approach," *CISS'99*, pp. 119 – 124, Mar. 1999.

[5] A. Mantravadi and V. Veeravalli, "On Discrete Sufficient Statistics for Asynchronous Band-limited CDMA Systems," in *Technical Report*, 1998, vol. No. 101.98.

[6] J. Holtzman, "A Simple, Accurate Method to Calculate SSMA Error Probability," *IEEE Trans. Communication*, vol. 49(3), pp. 461–464, March 1992.

[7] R. Morrow, "Accurate CDMA BER Calculation with Low Computational Complexity," in *IEEE Trans. on Communications*, November 1998, vol. 46(11), pp. 1114–1417.

[8] Viterbi, *CDMA: Principles of Spread Spectrum Communications*, Addison-Wesley Publishing Company, 1995.

[9] A. Viterbi and R. Padovani, "Implication of Mobile Cellular CDMA," in *IEEE Communications Magazine*, December 1992, pp. 38–41.

[10] R. Padovani, "Reverse Link Performance of IS-95 Based Cellular Systems," in *IEEE Personal Communications Magazine*, 3rd Quarter 1994, vol. 1, pp. 28–34.

[11] G. Stuber, *Principles of Mobile Communication*, Kluwer Academic Publishers, 1996.

[12] M. Dimentberg, *Statistical Dynamics of Nonlinear and Time Varying Systems*, John Wiley & Sons Inc., 1988.

[13] A. Fuller, *Nonlinear Stochastic Control Systems*, Barnes & Noble, New York, 1970.

[14] A. Pervozvanskii, *Random Process in Nonlinear Control System*, Academic Press, New York, 1965.

[15] C. Chen, *Linear System Theory and Design*, HBJ, Inc., Orlando, FL, 1984.

[16] Rec. ITU-R M.1225, *Guidelines for Evaluation of Radio Transmission Technologies for IMT-2000*, ITU, 1997.

A Class of Distributed Asynchronous Power Control Algorithms for Cellular Wireless Systems*

Jeffrey D. Herdtner and Edwin K. P. Chong
School of Electrical and Computer Engineering
Purdue University, West Lafayette, IN 47907-1285
{herdtner, echong}@ecn.purdue.edu

Abstract

We consider a class of distributed asynchronous power control algorithms based on the schemes used in IS-95 inner loop power control. Each user's received SIR is measured (using possibly outdated information) and compared to a threshold, and a single control bit is then sent to the user, indicating whether its power level should be increased or decreased. The SIR measurements and power updates do not require synchronization. We show that under certain conditions, this class of algorithms is stable and converges to a region around the optimal power assignment. We characterize this region and show that it can be made as small as desired by choosing algorithm parameters appropriately. For an appropriate choice of algorithm parameters, we show that convergence occurs in a finite number of iterations and derive an upper bound. The general results are illustrated with an example.

1 Introduction

We consider a wireless communication system consisting of mobile users communicating to base stations over a common radio channel. Because users are transmitting simultaneously over a shared channel, each user creates interference for the others. The goals of power control are to limit unnecessary interference and compensate for the effects of fading, thereby increasing battery life and improving quality of service.

Suppose mobile users have specific SIR requirements that may vary from user to user, possibly representing different classes of service. One goal of power control is to find a *feasible* power assignment, i.e., one that meets all users SIR constraints. Of course, this may not be possible, depending on the number of users, link gains, and other system parameters. We assume that some type of call admission control prevents the system from allowing such an infeasibility condition to arise. If a feasible assignment does exist, it has been shown that there is an optimal feasible power assignment, in the sense of minimizing the transmitter power levels of all users (see, e.g., [3]). One goal of power control is to determine this optimal power assignment in a distributed, asynchronous fashion.

Previous work that takes the above approach includes [3, 4, 9, 10]. Like the algorithm studied in this work, the algorithms presented there can be implemented in an asynchronous manner. The algorithm analyzed in our work differs mainly in that mobile

---

*This research supported in part by the National Science Foundation through grant ECS-9501652.

power adjustments require only a single bit of information to be transmitted to a user from its assigned base station. The bit simply indicates whether the user's SIR is above or below its desired level, and the mobile decreases (by a fixed amount α, in dB) or increases (by a fixed amount β, in dB) its power level accordingly.

The only previous analytical studies of this type of power control algorithm appear in [2] and [6]. (for simulation studies, see [1, 7, 8]). In [2], the authors establish the existence of step sizes ($\alpha = 1/\beta$) small enough to ensure stability and convergence (to within acceptable errors) of a synchronous version of the algorithm. We consider the asynchronous case with given fixed step sizes and characterize stability and convergence error in terms of the step sizes and system parameters such as the number of users and the users' SIR requirements. Our performance characterization gives insight into choosing step sizes that result in acceptable convergence errors and convergence speed.

The algorithm considered in [6] belongs to the class of algorithms considered in this work. Therefore, this work generalizes and extends the work in [6], which considered only the synchronous case with error-free transmission of power control commands.

Omitted proofs may be obtained by contacting the authors.

2 The Power Control Problem

Consider a wireless system with M base stations and N users communicating over a common radio channel. For $i = 1, 2, \ldots, N$, let $p_i \geq 0$ be the transmit power of user i, and let $\boldsymbol{p} = [p_1, \ldots, p_N]^T \geq \boldsymbol{0}$ be the corresponding power assignment vector (interpret vector inequalities componentwise). Users must transmit with enough power to overcome interference from noise and other users. Given a power assignment $\boldsymbol{p}$, let $I_i(\boldsymbol{p})$ be the interference plus noise that user i must overcome in order to meet its SIR requirement. Further, let $\boldsymbol{I}(\boldsymbol{p}) = [I_1(\boldsymbol{p}), \ldots, I_N(\boldsymbol{p})]^T$, and assume that $\boldsymbol{I}$ is a *standard interference function* as defined in [9]. A standard interference function satisfies the following three properties for power assignment vectors $\boldsymbol{p}$ and $\boldsymbol{p}'$:

(i) $\boldsymbol{I}(\boldsymbol{p}) > \boldsymbol{0}$ (positivity).

(ii) If $\boldsymbol{p} \geq \boldsymbol{p}'$, then $\boldsymbol{I}(\boldsymbol{p}) \geq \boldsymbol{I}(\boldsymbol{p}')$ (monotonicity).

(iii) If $c > 1$, then $c\boldsymbol{I}(\boldsymbol{p}) > \boldsymbol{I}(c\boldsymbol{p})$ (scalability).

See [9] for many examples of standard interference functions (we consider one example in Section 8). Because the channel is shared, we further assume that if any user's power level is increased without bound, then all other users are subject to a corresponding unbounded increase in interference. In other words, we assume that for each i and $j \neq i$, $I_j(\boldsymbol{p}) \to \infty$ as $p_i \to \infty$.

The goal of power control is to find the componentwise smallest (i.e., Pareto-minimal) power assignment vector that provides all users with acceptable connections. User i's connection is considered acceptable under power assignment $\boldsymbol{p}$ if $p_i \geq I_i(\boldsymbol{p})$. Therefore, the *power control problem* is to minimize $\boldsymbol{p}$ subject to $\boldsymbol{p} \geq \boldsymbol{I}(\boldsymbol{p})$, where the minimization is with respect to the partial order associated with componentwise vector inequality. Previous work using this formulation of the power control problem (or special cases) includes [3, 4, 9, 10]. The power control problem is called *feasible* if there is a power assignment vector $\boldsymbol{p}$ satisfying $\boldsymbol{p} \geq \boldsymbol{I}(\boldsymbol{p})$, i.e., if there is some power assignment for which all users have acceptable connections.

3 Standard Interference Functions

In this section, we give several important properties of standard interference functions (for proofs, contact the authors), beginning with continuity. Intuitively, continuity ensures that small changes in power levels do not result in large jumps in interference levels. It is natural to assume, as in [2], that the interference function for any real system is continuous, but it can actually be proved in a straightforward manner. Continuity is particularly important here, because, as will be shown later, the power control algorithms considered here do not converge exactly to the optimal power assignment. However, by appropriate choice of algorithm parameters, the convergence error can be made as small as desired, and continuity of the interference function ensures that the resulting connection quality is close to the desired level.

Lemma 1 *If* $\boldsymbol{p} > \mathbf{0}$*, then* $\boldsymbol{I}$ *is continuous at* $\boldsymbol{p}$*.*

An interference function $\boldsymbol{I}$ is called *feasible* if there exists a power assignment $\boldsymbol{p}$ such that $\boldsymbol{p} \geq \boldsymbol{I}(\boldsymbol{p})$ (i.e., if the associated power control problem is feasible). The next lemma states that a feasible standard interference function has a unique fixed point that is the solution to the associated power control problem. We omit the proof because it follows easily from [9].

Lemma 2 *If* $\boldsymbol{I}$ *is feasible, then* $\boldsymbol{I}$ *has a unique fixed point* $\boldsymbol{p}^* > \mathbf{0}$*, and* $\boldsymbol{p}^*$ *is the solution to the associated power control problem.*

We now define the *feasibility index*, R_I, of a standard interference function $\boldsymbol{I}$, which gives some measure of the size of the feasible region. Later, it will become clear that this quantity is related to how difficult it is for the algorithm described in Section 4 to solve the power control problem. The lemma following the definition illustrates the relationship between the feasibility index of a standard interference function and the feasibility of the associated power control problem.

Definition 1 *The feasibility index,* R_I*, of a standard interference function* $\boldsymbol{I}$ *is*

$$R_I \triangleq \sup\left\{c \in \mathbb{R} : \boldsymbol{p} \geq c\boldsymbol{I}(\boldsymbol{p}) \text{ for some power assignment } \boldsymbol{p}\right\}.$$

Lemma 3 $\boldsymbol{I}$ *is feasible if and only if* $R_I > 1$*.*

We now generalize Lemma 2 by showing that if $x \geq 0$ is small enough, then there is a componentwise minimum power assignment $\boldsymbol{p}$ such that $\boldsymbol{p} \geq x\boldsymbol{I}(\boldsymbol{p})$. In particular, if $0 \leq x < R_I$, then there is a componentwise minimum power assignment $\boldsymbol{p}$ such that $\boldsymbol{p} \geq x\boldsymbol{I}(\boldsymbol{p})$. The ratio $p_i/I_i(\boldsymbol{p})$ can be thought of as a measure of the quality of user i's connection, so the following proposition gives an upper bound on the connection quality simultaneously achievable by all users.

Proposition 1 *If* $x \in [0, R_I)$*, then there is a unique componentwise minimal power vector* $\tilde{\boldsymbol{p}}$ *such that* $\tilde{\boldsymbol{p}} \geq x\boldsymbol{I}(\tilde{\boldsymbol{p}})$*.*

Proof: Suppose $x \in (0, R_I)$ (the case when $x = 0$ is trivial), and let $\tilde{\boldsymbol{I}}(\boldsymbol{p}) = x\boldsymbol{I}(\boldsymbol{p})$. Then $\tilde{\boldsymbol{I}}$ is a standard interference function and $R_{\tilde{I}} = R_I/x > 1$, so Lemma 3 implies that $\tilde{\boldsymbol{I}}$ is feasible. Lemma 2 implies that $\tilde{\boldsymbol{I}}$ has a unique fixed point $\tilde{\boldsymbol{p}}$, which is the componentwise minimum vector satisfying $\boldsymbol{p} \geq \mathbf{0}$ and $\boldsymbol{p} \geq \tilde{\boldsymbol{I}}(\boldsymbol{p}) = x\boldsymbol{I}(\boldsymbol{p})$. ■

From the proof of this proposition, it is clear that $\boldsymbol{p} = x\boldsymbol{I}(\boldsymbol{p})$ is the componentwise minimum power vector satisfying $\boldsymbol{p} \geq x\boldsymbol{I}(\boldsymbol{p})$. For convenience, we denote this vector by $\overline{\boldsymbol{p}}(x)$. Note that $\overline{\boldsymbol{p}}(x)$ is defined for $x \in [0, R_I)$, and $\overline{\boldsymbol{p}}(x)$ is the componentwise smallest vector for which each user's power level is a fraction x of its desired value. If $R_I > 1$, then $\overline{\boldsymbol{p}}(1)$ exists and is the solution to the power control problem. Finally, note that each component of $\overline{\boldsymbol{p}}(x)$ is strictly monotone increasing in x.

4 Power Control Algorithm Description

We now describe the power control algorithm of interest. Let $N = \{0, 1, 2, \ldots\}$ be a set of times at which one or more users update their power levels. (The set N can be thought of as a set of indices of the physical times when power updates actually occur.) Let $N^i \subset N$ be the set of times at which user i's power level is updated. At times in N^i, user i's power level is increased by a factor $\beta > 1$ if its power level is too low to overcome the interference; otherwise, the power level is decreased by a factor α, where $0 < \alpha < 1$. We assume that N^i is an infinite set for all i, which implies that no user ever stops updating its power level. The algorithm is asynchronous because users may update their power levels at different times and with outdated information about other users' power levels. At time n, let $p_j\left(m_j^i(n)\right)$ be the value of user j's power level used in computing the interference seen by user i, where $m_j^i(n) \leq n$. Further, let

$$\boldsymbol{p}^i(n) \triangleq \left[p_1\left(m_1^i(n)\right), \ldots, p_N\left(m_N^i(n)\right)\right]^T$$

be the power vector used to compute the interference seen by user i at time n. Therefore, outdated measurements may be used in computing the interference, but we assume the measurements are always less than d iterations old, where $d \geq 1$. Therefore, $m_j^i(n) \geq n - (d-1)$ for all i, j, n. Given an initial power assignment $\boldsymbol{p}(0)$, the algorithm generates the sequence of power vectors $\{\boldsymbol{p}(n)\}_{n \in N}$ as follows:

$$p_i(n+1) = \begin{cases} \beta p_i(n) & \text{if } n \in N^i \text{ and } p_i^i(n) < I^i(n) \\ \alpha p_i(n) & \text{if } n \in N^i \text{ and } p_i^i(n) \geq I^i(n) \\ p_i(n) & \text{if } n \notin N^i \end{cases}, \tag{1}$$

where $n \in N$, and $I^i(n) \triangleq I_i(\boldsymbol{p}^i(n))$ is the most recent measured value (at time n) of the interference seen by user i. This algorithm has very desirable characteristics. No global information or synchronization is needed for users to update their power levels. To update its power level, each user needs only a single bit of information from its assigned base station, indicating whether or not its connection is acceptable. Each user then either increases its power level by a factor β or decreases it by a factor α. (Note that this is equivalent to increasing or decreasing the power level by a fixed amount in dB.) It is assumed that the base stations always accurately determine whether or not connections are acceptable and that the corresponding information bit is transmitted without error to the user. However, in less ideal situations, a special case of this type of algorithm was shown to be robust in simulations (see [1, 8]).

In Section 5, we show that this algorithm is stable whenever $\alpha v^d < R_I$, where $v = \beta/\alpha$. It is impossible for this algorithm to converge exactly to the optimal power assignment, because the jump sizes do not approach zero as the algorithm progresses. However, it may come close to converging, and this is shown in Section 6.

5 Stability

An algorithm is considered to be stable if, for any initial condition $\boldsymbol{p}(0)$, there is a finite power vector $\overline{\boldsymbol{p}}$ such that $\boldsymbol{p}(n) \leq \overline{\boldsymbol{p}}$ for all $n \geq 0$. To this end, the following lemma gives a characterization of a region of power assignments that is invariant under the power control iteration (1), provided $\alpha v^d < R_I$. This result is useful in proving the stability of the power control algorithm. For notational convenience, let $C(\overline{\boldsymbol{p}}) \triangleq \{\boldsymbol{p} \in \mathbb{R}^N : \boldsymbol{p} \leq \overline{\boldsymbol{p}}\}$.

Lemma 4 *Suppose $\alpha v^d < R_I$. If $x \in [\alpha v^d, R_I)$, then $C(\overline{\boldsymbol{p}}(x))$ is invariant under iteration (1).*

Note that the invariant region is written in terms of $\overline{\boldsymbol{p}}(x)$, but the actual power assignments can be any nonnegative vector $\boldsymbol{p}$. If we can guarantee that, for any $\boldsymbol{p} \geq \boldsymbol{0}$, there is some number x such that $\overline{\boldsymbol{p}}(x) \geq \boldsymbol{p}$, then stability will follow easily from the invariance lemma above. Therefore, the following lemma is very useful.

Lemma 5 *For all i, $\overline{p}_i(x) \nearrow \infty$ as $x \nearrow R_I$.*

In light of the lemma, we are justified in defining $z(\boldsymbol{p})$ to be the smallest number x such that $\overline{\boldsymbol{p}}(x) \geq \boldsymbol{p}$. In other words, for any power vector $\boldsymbol{p}$, we let $z(\boldsymbol{p}) \triangleq \min\{0 \leq x < R_I : \overline{\boldsymbol{p}}(x) \geq \boldsymbol{p}\}$. It is now simple to prove that the power control algorithm defined by (1) is stable if $\alpha v^d < R_I$.

Theorem 1 *If $\alpha v^d < R_I$, then the iteration (1) is stable.*

Proof: Suppose $\alpha v^d < R_I$, and let $\boldsymbol{p}(0) \geq \boldsymbol{0}$ be the initial power assignment. Let $z_0 = z(\boldsymbol{p}(0))$, $x = \max\{\alpha v^d, z_0\}$, and $\overline{\boldsymbol{p}} = \overline{\boldsymbol{p}}(x)$. By the definitions of z and $\overline{\boldsymbol{p}}(\cdot)$, we have $\boldsymbol{p}(0) \leq \overline{\boldsymbol{p}}(z_0) \leq \overline{\boldsymbol{p}}(x) = \overline{\boldsymbol{p}}$, so $\boldsymbol{p}(0) \in C(\overline{\boldsymbol{p}})$.

Now suppose $\boldsymbol{p}(n) \in C(\overline{\boldsymbol{p}})$ for some $n \geq 0$. Since $\alpha v^d \leq x < R_I$, Lemma 4 implies that $\boldsymbol{p}(n+1) \in C(\overline{\boldsymbol{p}})$. Therefore, by induction, $\boldsymbol{p}(n) \in C(\overline{\boldsymbol{p}})$ for all $n \geq 0$, which completes the proof. ■

We have proved stability of the power control algorithm, but this gives little information about how close the algorithm comes to the optimal power assignment. The bound in the above proof is dependent on the initial power assignment, which is undesirable. In Section 6, we find a tighter bound that does not depend on the initial condition.

6 Power Level and SIR Convergence

In Section 3, it was established that if $R_I > 1$, then there is a componentwise minimum power assignment, namely $\overline{\boldsymbol{p}}(1)$, which provides all users with acceptable connections. In this section, we show that $\limsup \boldsymbol{p}(n) \leq \overline{\boldsymbol{p}}(\alpha v^d)$ and $\liminf \boldsymbol{p}(n) \geq \overline{\boldsymbol{p}}(\beta v^{-d})$. Since $\overline{\boldsymbol{p}}(\cdot)$ is continuous, $\overline{\boldsymbol{p}}(\alpha v^d) \to \boldsymbol{p}^*$ and $\overline{\boldsymbol{p}}(\beta v^{-d}) \to \boldsymbol{p}^*$ as $\alpha, \beta \to 1$. Therefore, the power assignment error can be controlled by choosing β and α close to 1.

Theorem 2 *If $\alpha v^d < R_I$, then $\limsup \boldsymbol{p}(n) \leq \overline{\boldsymbol{p}}(v^d)$ and $\liminf \boldsymbol{p}(n) \geq \overline{\boldsymbol{p}}(v^{-d})$.*

Recall that a user's connection is considered acceptable if $p_i(n)/I_i(\boldsymbol{p}(n)) \geq 1$. Convergence to nearly optimal power levels also guarantees nearly acceptable connections for all users. Eventual lower and upper bounds on the ratio $p_i(n)/I_i(\boldsymbol{p}(n))$ can be expressed in terms of the power level bounds. However, for the synchronous version of the algorithm, we now derive much simpler bounds, slightly generalizing a result in [2]. In particular, we show that the ratio $p_i(n)/I_i(\boldsymbol{p}(n)) \geq (\alpha/\beta)^d$ after a finite number of iterations, and never drops below thereafter. For example, if $d = 1$ and a connection is considered acceptable when $p_i(n)$ is within 1 dB of $I_i(\boldsymbol{p}(n))$, then one way to guarantee that all users have acceptable connections after a finite number of iterations is to set $\alpha = -0.5$ dB and $\beta = 0.5$ dB.

Note that, although this result is only for the synchronous case, it does not mean all power updates must take place simultaneously. Here, we use the term synchronous to mean that the measurements of the users' connection qualities at the base stations must be made simultaneously. To the mobiles, synchronicity only means that all users must update their power levels at the same rate, thus resulting in synchronicity in the discrete-time domain but not necessarily synchronicity in real time. In other words, in the notation of Section 4, we require that $\boldsymbol{p}^i(n) = \boldsymbol{p}(n-(d-1))$ for all i and thus that $I^i(n) = I_i(\boldsymbol{p}(n-(d-1)))$ for all i.

Proposition 2 *Consider the synchronous case of the power control algorithm defined by (1). If the algorithm is stable, then there exists a nonnegative integer K such that $(\alpha/\beta)^d \leq p_i(n)/I_i(\boldsymbol{p}(n)) \leq (\beta/\alpha)^d$ for all $n \geq K$.*

Proof: We prove only the lower bound (the proof of the upper bound is similar). For convenience, let $y_i(n) = p_i(n)/I_i(\boldsymbol{p}(n))$ and $\boldsymbol{y}(n) = [y_1(n), \ldots, y_N(n)]^T$. Suppose $y_i(n-1) \geq (\alpha/\beta)^d$ for some user i and some $n \geq 1$. If $y_i(n-d) < 1$, then

$$y_i(n) = \frac{\beta p_i(n-1)}{I_i(\boldsymbol{p}(n))} \geq \frac{\beta p_i(n-1)}{I_i(\beta \boldsymbol{p}(n-1))} > y_i(n-1) \geq \left(\frac{\alpha}{\beta}\right)^d,$$

and if $y_i(n-d) \geq 1$, then

$$y_i(n) \geq \frac{\alpha^d p_i(n-d)}{I_i(\beta^d \boldsymbol{p}(n-d))} > \frac{\alpha^d p_i(n-d)}{\beta^d I_i(\boldsymbol{p}(n-d))} = \left(\frac{\alpha}{\beta}\right)^d y_i(n-d) \geq \left(\frac{\alpha}{\beta}\right)^d.$$

Therefore, if $y_i(k) \geq (\alpha/\beta)^d$ at some iteration k, then $y_i(n) \geq (\alpha/\beta)^d$ for all $n \geq k$.

It remains to show that for each i, there exists a nonnegative integer K_i such that $y_i(K_i) \geq (\alpha/\beta)^d$. Suppose not, i.e., suppose for some i, we have $y_i(n) < (\alpha/\beta)^d$ for all $n \geq 0$. Then for $n \geq d-1$, $p_i(n) = \beta^{n-(d-1)} p_i(0) \to \infty$ as $n \to \infty$, which contradicts algorithm stability. Therefore, for each i, there exists a nonnegative integer K_i such that $y_i(K_i) \geq (\alpha/\beta)^d$. Finally, $\boldsymbol{y}(n) \geq (\alpha/\beta)^d$ for all $n \geq K = \max(K_1, \ldots, K_N)$, which completes the proof. ■

7 Convergence Speed

Power control will ultimately be applied in a dynamic setting, with possibly rapidly changing conditions, so it is important to study the speed at which the power control algorithm converges. Although we study the convergence rate only in the static setting, the algorithm will still perform well in a dynamic setting if it converges faster than the conditions change. For further justification of this "separation of time scales" argument in large systems, see [5]. In practice, the algorithm may not always be able to compensate completely for fast fading effects, and some other mechanism must be used. For example, in IS-95, if poor performance is detected, the target SIR is adjusted by an "outer loop" power control mechanism.

Although the algorithm presented here does not converge to a single point, we have shown that it converges to a region about the optimal power assignment. More specifically, we have shown that the algorithm either reaches the convergence region in a finite number of steps (and remains there) or approaches the convergence region. In fact, in the special case where $\alpha = 1/\beta$, the synchronous version of the algorithm always reaches the convergence region in a finite number of iterations. We prove this for the case where $d = 1$ below. For general d, the proof is similar.

Proposition 3 *Consider the synchronous version of the power control algorithm, and let* $\alpha = 1/\beta$. *If* $R_I > \alpha v^d$, *then there is a finite integer* L *such that* $\overline{\boldsymbol{p}}(\beta v^{-d}) \leq \boldsymbol{p}(n) \leq \alpha\overline{\boldsymbol{p}}(\alpha v^d)$ *for all* $n > L$.

Proof: Assume $d = 1$ (for $d > 1$, the proof is similar). We will show that there is a finite integer L such that $\boldsymbol{p}(n) \leq \overline{\boldsymbol{p}}(\beta)$ for all $n > L$ (showing the lower bound is similar). Let $z_n = z(\boldsymbol{p}(n))$. Lemma 4 implies that it is only necessary to find some integer n such that $\boldsymbol{p}(n) \leq \overline{\boldsymbol{p}}(\beta)$, so the case where $z_0 \leq \beta$ is trivial.

Assume $z_0 > \beta$. Note that synchronicity is equivalent to letting $N_i = N$ for all i. Because $\alpha = 1/\beta$, user i's power level at time n must satisfy $p_i(n) = \beta^k p_i(0)$ for some integer k. If $z_n > \beta$, then it can be shown that $z_{n+1} < z_n$, which in turn implies that $\overline{\boldsymbol{p}}(z_{n+1}) < \overline{\boldsymbol{p}}(z_n)$. Let m_i be the number of possible power assignments $p_i(n)$ satisfying $\overline{p}_i(\beta) < p_i(n) \leq \overline{p}_i(z_0)$. Then, we have

$$m_i \leq \left\lceil \log_\beta \frac{\overline{p}_i(z_0)}{\overline{p}_i(\beta)} \right\rceil .$$

Each element of the sequence $\{\overline{\boldsymbol{p}}(z_n)\}$ satisfies $\overline{p}_i(z_n) = p_i(n)$ for some i. Since $\overline{\boldsymbol{p}}(z_n)$ strictly decreases when $z_n > \beta$ (i.e., when $p_i(n) > \overline{p}_i(\beta)$ for some i), there are at most L elements of the sequence $\{\overline{\boldsymbol{p}}(z_n)\}$ satisfying $\overline{\boldsymbol{p}}(z_n) \geq \overline{\boldsymbol{p}}(\beta)$, where

$$L = \sum_{i=1}^{N} m_i \leq \sum_{i=1}^{N} \left\lceil \log_\beta \frac{\overline{p}_i(z_0)}{\overline{p}_i(\beta)} \right\rceil .$$

Therefore, $\overline{\boldsymbol{p}}(z_n) \leq \overline{\boldsymbol{p}}(\beta)$ for all $n > L$, so $\boldsymbol{p}(n) \leq \overline{\boldsymbol{p}}(z_n) \leq \overline{\boldsymbol{p}}(\beta)$ for all $n > L$. ∎

8 Application to Fixed Base Station Assignment

We now illustrate the application of the preceding results to systems with fixed base station assignment. In cellular systems, users are sometimes assigned to communicate to a base station based on signal strength, location, or some other parameter independent of the power control algorithm. From the perspective of power control, this is called fixed base station assignment (although it does not preclude handoffs). In a CDMA system with fixed base station assignment, it is common to model the SIR of user i as

$$SIR_i(\boldsymbol{p}) = \frac{h_{ii}p_i}{\sum_{j \neq i} h_{ji}p_j + \sigma_i},$$

where $h_{ij} > 0$ is the "gain" from user i to the assigned base station of user j, and $\sigma_i > 0$ is the receiver noise at the assigned base station of user i. Assume that user i has a target SIR of $\gamma_i > 0$, i.e., it is desired that $SIR_i(\boldsymbol{p}) \geq \gamma_i$ for each user i. This is equivalent to $p_i \geq I_i(\boldsymbol{p})$, where

$$I_i(\boldsymbol{p}) \triangleq \frac{\gamma_i}{h_{ii}}\left(\sum_{j \neq i} h_{ji}p_j + \sigma_i\right). \tag{2}$$

It is easy to verify that $\boldsymbol{I}(\boldsymbol{p}) = [I_1(\boldsymbol{p}), \ldots, I_N(\boldsymbol{p})]^T$ is a standard interference function, and for each i and $j \neq i$, $I_j(\boldsymbol{p}) \to \infty$ as $p_i \to \infty$.

From iteration (1), implementing the fixed base station assignment algorithm requires comparing $p_i^i(n)$ to $I^i(n)$. By letting $r_i(n) = h_{ii}p_i^i(n)$ and $r_i^{\text{tot}}(n) = \sum_{j \neq i} h_{ji}p_j^i(n) + \sigma_i$ be the most recent measured values (at the assigned base station of user i) of the received power of user i and the total received power plus noise, respectively, it is easy to show that the condition $p_i^i(n) < I^i(n)$ is equivalent to

$$\frac{r_i(n)}{r_i^{\text{tot}}(n)} < \frac{\gamma_i}{1+\gamma_i}.$$

Therefore, unlike many power control algorithms proposed in the literature, implementation of this algorithm does not require any measurements of users' link gains.

Since the interference function given in (2) is standard, we can apply the results derived in previous sections. Those results depend on the feasibility index, R_I, of the standard interference function. To calculate R_I for the interference function given by (2), first note that the condition $\boldsymbol{p} \geq \boldsymbol{I}(\boldsymbol{p})$ can be written in matrix form by letting $\boldsymbol{H}$ be the $N \times N$ *normalized gain matrix* with entries $[\boldsymbol{H}]_{ij} = \gamma_i h_{ji}/h_{ii}$ for $j \neq i$ and $[\boldsymbol{H}]_{ij} = 0$ for $j = i$, and letting $\boldsymbol{\delta}$ be the N-vector with entries $\delta_i = \gamma_i\sigma_i/h_{ii}$. Then $\boldsymbol{I}(\boldsymbol{p}) = \boldsymbol{H}\boldsymbol{p} + \boldsymbol{\delta}$, so the condition $\boldsymbol{p} \geq \boldsymbol{I}(\boldsymbol{p})$ is equivalent to $\boldsymbol{p} \geq \boldsymbol{H}\boldsymbol{p} + \boldsymbol{\delta}$.

Let $\lambda_{\max}$ denote the spectral radius of $\boldsymbol{H}$. The following theorem establishes the relationship between the feasibility index of $\boldsymbol{I}$ and the spectral radius of $\boldsymbol{H}$.

Theorem 3 *Let $\boldsymbol{I}$ be as defined in (2). Then $R_I = 1/\lambda_{max}$.*

It is expected that the feasibility index will decrease as the number of users in the system increases. At some point, the feasibility index will drop below one, meaning that not all SIR targets can be met. This is illustrated in Figure 1. The data was generated by simulating a system of mobile users randomly positioned about ten base stations on a ring (for simplicity). The gains h_{ij} from the users to the base stations were assumed to consist of a fourth power propagation loss factor and log-normal shadowing with zero mean and a standard deviation of 8 dB. Users were assigned to the base station for which the gain was highest. The delay was set to $d = 1$, and the target SIR for all users was set at 1/20 (−13 dB). Systems of a given size (i.e., number of users) were generated repeatedly, and the corresponding feasibility indices were averaged and plotted. Therefore, the point where the feasibility index drops below unity does not represent a hard limit on the number of feasible users. In other words, given a certain number of users, it is not possible to know for sure if they can all be accommodated. This complicates the call admission control process. However, it may still happen that below some threshold, it is very probable that all users can be accommodated. Figure 1 illustrates that this is indeed true. The data was generated by repeatedly generating systems with a given number of users and computing the fraction of feasible systems.

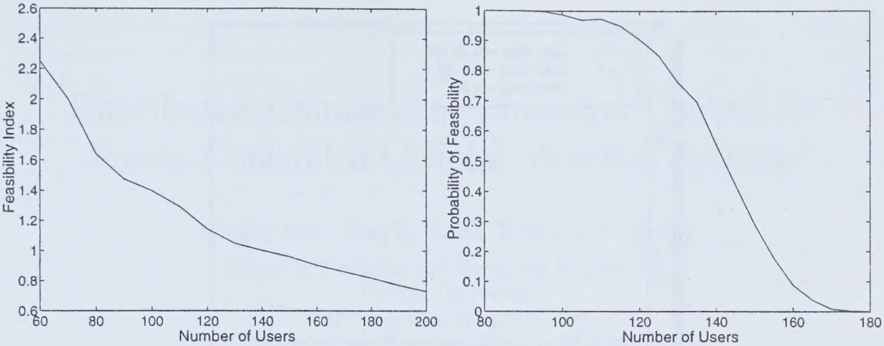


Figure 1: The feasibility index and probability of feasibility for systems with fixed base station assignment.

Recall that $\overline{\boldsymbol{p}}(x)$ is well-defined for $x \in [0, R_I)$, and $\overline{\boldsymbol{p}}(x)$ is the unique power assignment satisfying $\boldsymbol{p} = x\boldsymbol{I}(\boldsymbol{p})$. For the interference function given by (2), it is easy to show that $\overline{\boldsymbol{p}}(x)$ is given explicitly by $\overline{\boldsymbol{p}}(x) = x(\boldsymbol{I} - x\boldsymbol{H})^{-1}\boldsymbol{\delta}$, for $x \in [0, 1/\lambda_{\max})$. The following result now follows easily from Theorems 2 and 3.

Proposition 4 *Let $\{\boldsymbol{p}(n)\}$ be the sequence of power assignments generated by iteration (1), with $\boldsymbol{I}$ given by (2). If $\alpha v^d < 1/\lambda_{max}$, then*

$$\beta v^{-d}(\boldsymbol{I} - \beta v^{-d}\boldsymbol{H})^{-1}\boldsymbol{\delta} \leq \liminf \boldsymbol{p}(n) \leq \limsup \boldsymbol{p}(n) \leq \alpha v^d(\boldsymbol{I} - \alpha v^d \boldsymbol{H})^{-1}\boldsymbol{\delta}.$$

To illustrate the relationship between the bounds and the optimal power assignment, simulated system data was generated as described above. For a given number of users, many systems were generated, the upper bounds were computed, and the worst case deviations from the optimal power assignment were averaged. The results are plotted versus the number of users in Figure 2, parameterized by the algorithm step size, β (in all cases, α was constrained to satisfy $\alpha = 1/\beta$). The deviations are not too significant, except when the step size and number of users are large. In addition, keep in mind that this is the worst case user, and the bound is for instantaneous rather than average power. Because the power level will fluctuate upwards and downwards with time, the average power level will likely be reasonably close to the optimal level.

Additional e

References

[1] S. Ariyavisitakul, "Signal and interference statistics of a CDMA system with feedback power control - Part II," *IEEE Transactions on Communications*, vol. 42, pp. 597–605, 1994.

[2] P. J. Fleming and A. L. Stolyar, "Convergence properties of the IS-95 reverse link power control algorithm," manuscript, July 1996.

[3] G. J. Foschini and Z. Miljanic, "A simple distributed autonomous power control algorithm and its convergence," *IEEE Transactions on Vehicular Technology*, vol. 42, no. 4, pp. 641–646, Nov. 1993.

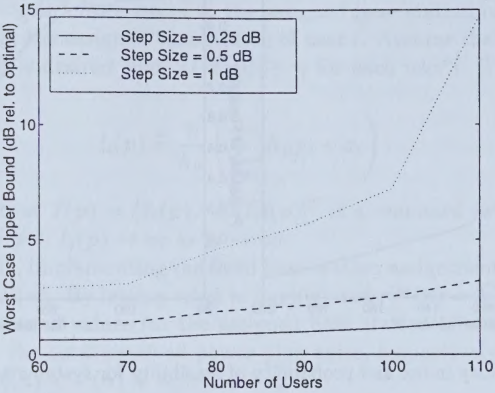


Figure 2: The worst case power level relative to the optimal power level under fixed base station assignment. Each data point represents the average over many systems.

[4] S. V. Hanly, "An algorithm for combined cell-site selection and power control to maximize cellular spread spectrum capacity," *IEEE Journal on Selected Areas in Communications*, vol. 13, no. 7, pp. 1332–1340, Sept. 1995.

[5] S. V. Hanly, "Capacity and power control in spread spectrum macrodiversity radio networks," *IEEE Transactions on Communications*, vol. 44, no. 2, pp. 247–256, Feb. 1996.

[6] J. D. Herdtner and E. K. P. Chong, "Analysis of a distributed power control algorithm," in *Proc. of 36th Annual Allerton Conference on Communication, Control, and Computing*, pp. 296–297, Sept. 1998.

[7] F. Ling, B. Love, and M. M. Wang, "Behavior and performance of power controlled IS-95 reverse-link under soft handoff," in *Proc. IEEE Vehicular Technology Conference*, vol. 2, pp. 924–928, Sept. 1997.

[8] A. J. Viterbi, A. M. Viterbi, and E. Zehavi, "Performance of power-controlled wideband terrestrial digital communication," *IEEE Transactions on Communications*, vol. 41, no. 4, pp. 559–569, Apr. 1993.

[9] R. D. Yates, "A framework for uplink power control in cellular radio systems," *IEEE Journal on Selected Areas in Communications*, vol. 13, no. 7, pp. 1341–1347, Sept. 1995.

[10] R. D. Yates and C.-Y. Huang, "Integrated power control and base station assignment," *IEEE Transactions on Vehicular Technology*, vol. 44, no. 3, pp. 638–644, Aug. 1995.

Distributed Connection Admission Control for Power-Controlled Cellular Wireless Systems*

Mingbo Xiao, Ness B. Shroff, Edwin K. P. Chong
School of Electrical and Computer Engineering
Purdue University
West Lafayette, IN 47907-1285
{mingbo, shroff, echong}@ecn.purdue.edu

Abstract

It is well-known that power control can help to improve spectrum utilization in cellular wireless systems. However, many existing distributed power control algorithms do not work well without an effective connection admission control mechanism, because they could diverge and result in dropping existing calls when an infeasible call is admitted. In this work, based on a system parameter defined as the discriminant, we propose two distributed CAC algorithms for a power-controlled system. Under these CAC schemes, an infeasible call is rejected early, and incurs only a small disturbance to existing calls, while a feasible call is admitted and the system converges to the Pareto optimal power assignment.

1 Introduction

The tremendous success of cellular phones has generated great interest in wireless networks. Wireless subscribers have now begun to expect many advanced networking capabilities, such as multimedia applications, multicasting, and guaranteed quality of service (QoS). However, the wireless network is characterized by scarce radio spectrum, an unreliable propagation channel (with shadowing, multipath fading, etc.), and node mobility, all of which lead to a number of interesting open problems for network management in these systems [1]. In this paper, we address one such important problem: connection admission control (CAC).

The goal of an efficient CAC scheme is to guarantee the quality of service of the ongoing connections, while at the same time efficiently using the available radio spectrum. Connection admission control in the form of bang-bang control (admission or rejection) by itself is not effective to control and guarantee QoS. This is especially true when a cellular system adopts fixed channel allocation (FCA) with neither power control nor mobility prediction. However, armed with dynamic channel allocation (DCA), power control, and mobility prediction, CAC has many more degrees of freedom, and can be very useful in guaranteeing QoS. Our goal is to design an efficient CAC scheme that is efficient, simple, and robust enough to implement.

Power control in a cellular system has been shown to have numerous benefits. It reduces co-channel interference and guarantees the signal to interference ratio (SIR) of

---

*This research was supported in part by the National Science Foundation through grants ECS-9501652, NCR-9624525, and ANI-9805441.

ongoing connections, resulting in a higher utilization and/or better QoS. If power levels are tuned continuously, power control can be a strong enhancement to the bang-bang type of call admission control. There have been numerous papers on power control, especially for CDMA systems. From the viewpoint of practical applications, distributed power control schemes are of special interest and importance. In [7], Yates unifies most of the known distributed power control schemes under a framework called *standard power control.* Every algorithm under this framework converges for both synchronous and asynchronous cases, when the system is feasible. However, if there is no feasible power assignment, a distributed power control algorithm can diverge or result in dropping an existing call. Thus, a criterion is needed to decide if a set of mobiles in the system can be served at the same time. A well-known criterion can be found in [2, 6], but it is centralized and is not suitable for distributed implementation.

In this paper, we will define a system parameter called the discriminant. Based on it, we will present a criterion that can decide, in a distributed way, the feasibility of a new call arriving to a system with a set of feasible calls in service. The resulting CAC algorithms are independent of the power control schemes and multiple access techniques used. They are measurement-based, and assume nothing about the traffic statistics. They are optimal in that they admit feasible calls and guarantee system convergence to the Pareto optimal power assignment (to be described later), while rejecting infeasible calls.

The rest of the paper is organized as follows. In Section 2, we first present the system model, and then some existing results (mainly from [2, 3]). In Section 3, we define the discriminant, describe the admissibility conditions, and propose distributed CAC mechanisms for power-controlled systems. Finally, the last section concludes the paper and discusses directions towards future work.

2 System Model and Some Existing Work

We consider a power-controlled cellular system where the transmitted powers are continuously tunable. In the system, every mobile is associated with a base station (called its home base station) to communicate. To maintain a connection between the mobile and its home base station, the signal to interference ratio (SIR) at the receiver must be no less than some threshold, which corresponds to the requirement of QoS such as the bit error rate. Since the uplink can be treated similarly to the downlink, in this paper we consider only downlink transmissions.

Let $M_n = \{m_1, m_2, \ldots, m_n\}$ be the set of n mobiles in the system, and let p_i be the downlink power level of mobile m_i. Let G_{ij} denote the gain from home base station of m_j to mobile m_i. Then the power received at mobile m_i from the downlink of m_j is $G_{ij}p_j$. Let η_i be the thermal noise received at mobile m_i, and let γ_i be its required SIR threshold. Then we need

$$SIR_i \equiv \frac{G_{ii}p_i}{\sum_{j\neq i} G_{ij}p_j + \eta_i} \geq \gamma_i, \tag{1}$$

to maintain the downlink connection for mobile m_i. Note that this model is general enough to represent DS-CDMA systems with matched-filter receiver [10] or one-channel TDMA/FDMA systems [2], by giving specific interpretations to the parameters.

We can rewrite (1) as $p_i - \sum_{j\neq i} \frac{\gamma_i G_{ij}}{G_{ii}} p_j \geq \frac{\gamma_i \eta_i}{G_{ii}}$. Let I_n be the $n \times n$ identity matrix, $P_n = [p_1, p_2, \ldots, p_n]^{\mathrm{T}}$, $U_n = [\frac{\gamma_1\eta_1}{G_{11}}, \frac{\gamma_2\eta_2}{G_{22}}, \ldots, \frac{\gamma_n\eta_n}{G_{nn}}]^{\mathrm{T}}$, and F_n be the $n \times n$ non-negative matrix with (i,j) entry $F_{ij} = \frac{\gamma_i G_{ij}}{G_{ii}}$, if $i \neq j$, otherwise 0. Then the power assignment P_n is *feasible* if

$$\begin{cases} (I_n - F_n)P_n \geq U_n \\ P_n > \vec{0}_n, \end{cases} \tag{2}$$

where $\vec{0}_n$ is the all-zero vector with dimension n, and the inequalities are componentwise. For the system we just described, we now state a theorem and two corollaries that directly follow from results in [3, 6].

Theorem 1 *The following statments are equivalent:*

1. *There exists a feasible power assignment* P_n.
2. *The maximum modulus eigenvalue (Perron-Frobenius eigenvalue) of* F_n *is smaller than 1.*
3. $(I_n - F_n)^{-1} = \sum_{k=0}^{\infty}(F_n)^k$ *exists and is componentwise positive.*

A power assignment P_n' is said to be *Pareto optimal* (or simply *optimal*) if it is feasible and any other feasible power assignment P_n satisfies $P_n \geq P_n'$ componentwise.

Corollary 1 *If* $P_n' = (I_n - F_n)^{-1}U_n$ *is not componentwise positive, then there is no feasible* P_n*; Otherwise,* P_n' *is Pareto optimal.*

Corollary 2 *If a Pareto optimal assignment* P_n' *exists, then the iterative power updating algorithm* $P_n(k+1) = F_nP_n(k) + U_n$*, or alternatively,*

$$p_i(k+1) = \frac{\gamma_i}{SIR_i} p_i(k), \text{ for } i = 1, \ldots, n, \tag{3}$$

converges (from arbitrary $P_n(0) > 0$*) to* P_n' *with geometric convergence.*

The algorithm (3) is distributed and autonomous because it relies only on local information. It is also asynchronously convergent [6]. However, this algorithm does not work well without an effective CAC, because it diverges and results in some existing calls being dropped if there is no feasible power assignment. A modified algorithm of (3), DPC-ALP, may also experience convergence problems, despite its slower convergence rate. The technique of probing [4, 5] and a similar method in [8] aim to overcome this problem, but they depend on approximations involving measurements of small quantities and may not be easy to implement.

In [2], it has been shown that the problem of finding the Pareto optimal power assignment P_n can also be reformulated as finding a basic feasible solution $[P_n^{\mathrm{T}}\ X_n^{\mathrm{T}}]^{\mathrm{T}}$, through Gaussian reduction, for the following set of linear equations:

$$\begin{cases} (I_n - F_n)P_n + I_nX_n = U_n \\ P_n^{\mathrm{T}}X_n = 0; P_n, X_n \geq \vec{0}_n. \end{cases} \tag{4}$$

The procedure can decide the feasible subset of mobiles in M_n, as well as the optimal power assignment. Though this algorithm is centralized, it gives us the insight that a mobile is admissible if and only if its pivoting variable in the Gaussian reduction tableau is positive. Based on this insight we next provide admissibility criteria and two distributed CAC algorithms.

3 Admissible Conditions and Distributed CAC algorithms

The role of admission control is to decide if a new call can be accepted; if yes, the problem then becomes one of using which channel (or signature code) and at what power level. In this section, we consider the case where n feasible calls (namely, $m_1, \ldots, m_n$) are in service, and a new call m_{n+1} arrives and tries to get admission. Denote the base station assigned to the new call by B_{n+1}. To achieve higher spectrum utilization, the power control algorithm used should converge to the Pareto optimal assignment, when it is feasible. We call a power control algorithm satisfying this condition *optimal.* Many existing power control algorithms (e.g., algorithm (3)) are optimal, but DPC-ALP [3] is not. We only consider the CAC problem for systems under optimal power control throughout the paper. We also assume that the existing system is initially in steady state.

3.1 Admissible Condition

Before admitting mobile m_{n+1}, there are n feasible calls in the system, with power assignment $P_{n,old}$. Hence, the following inequality holds:

$$\left\{ \begin{array}{l} (I_n - F_n)P_{n,old} \geq U_n \\ P_{n,old} > \vec{0}_n, \end{array} \right. \tag{5}$$

and the optimal power assignment is $P^*_{n,old} = (I_n - F_n)^{-1}U_n$. The newly arrived mobile m_{n+1} (with SIR threshold γ_{n+1}, and thermal noise η_{n+1}) is *admissible* if and only if there exists a power assignment P_{n+1} for the $(n+1)$ downlinks satisfying

$$\left\{ \begin{array}{l} (I_{n+1} - F_{n+1})P_{n+1} \geq U_{n+1} \\ P_{n+1} > \vec{0}_{n+1}. \end{array} \right. \tag{6}$$

To be specific, there exist positive power assignments P_n and p_{n+1} for the n existing mobiles and m_{n+1}, respectively, such that

$$\begin{bmatrix} I_n - F_n & -\vec{F}_{\cdot,n+1} \\ -\vec{F}^{\mathrm{T}}_{n+1,\cdot} & 1 \end{bmatrix} \begin{bmatrix} P_n \\ p_{n+1} \end{bmatrix} \geq \begin{bmatrix} U_n \\ u_{n+1} \end{bmatrix}, \tag{7}$$

where

$\vec{F}_{\cdot,n+1} = [F_{1,n+1}, F_{2,n+1}, \ldots, F_{n,n+1}]^{\mathrm{T}}$, with $F_{i,n+1} = \frac{\gamma_i G_{i,n+1}}{G_{ii}}$;

$\vec{F}_{n+1,\cdot} = [F_{n+1,1}, F_{n+1,2}, \ldots, F_{n+1,n}]^{\mathrm{T}}$, with $F_{n+1,i} = \frac{\gamma_{n+1} G_{n+1,i}}{G_{n+1,n+1}}$;

$u_{n+1} = \frac{\gamma_{n+1}\eta_{n+1}}{G_{n+1,n+1}}$.

The quantity $F_{i,n+1}$ corresponds to the interference from the downlink of m_{n+1} to mobile m_i, and $F_{n+1,i}$ from the downlink of m_i to mobile m_{n+1}.

Let $\Delta_{n+1} = 1 - \vec{F}^{\mathrm{T}}_{n+1,\cdot}(I_n - F_n)^{-1}\vec{F}_{\cdot,n+1}$. We call Δ_{n+1} the *discriminant of* m_{n+1}, because it provides a criterion for the admissibility of m_{n+1}. This criterion is given by the following theorem.

Theorem 2 *Mobile m_{n+1} is admissible if and only if $\Delta_{n+1} > 0$. Further, if m_{n+1} is admissible, then the optimal power assignment P^*_{n+1} is given by*

$$P^*_{n+1} = \begin{bmatrix} P^*_n \\ p^*_{n+1} \end{bmatrix} = \begin{bmatrix} P^*_{n,old} + (I_n - F_n)^{-1}\vec{F}_{\cdot,n+1}p^*_{n+1} \\ \frac{\gamma_{n+1}}{G_{n+1,n+1}\Delta_{n+1}} R^{\text{initial}}_{n+1} \end{bmatrix}, \tag{8}$$

where $R_{n+1}^{\text{initial}} = [G_{n+1,1}, G_{n+1,2}, \ldots, G_{n+1,n}]P_{n,old}^{*} + \eta_{n+1}$.

Proof: We have

$$
\begin{aligned}
(I_{n+1} - F_{n+1})^{-1} &= \begin{bmatrix} I_n - F_n & -\vec{F}_{\cdot,n+1} \\ -\vec{F}_{n+1,\cdot}^{\mathrm{T}} & 1 \end{bmatrix}^{-1} \\
&= \begin{bmatrix} (I_n - F_n)^{-1} + \frac{(I_n-F_n)^{-1}\vec{F}_{\cdot,n+1}\vec{F}_{n+1,\cdot}^{\mathrm{T}}(I_n-F_n)^{-1}}{\Delta_{n+1}} & \frac{(I_n-F_n)^{-1}\vec{F}_{\cdot,n+1}}{\Delta_{n+1}} \\ \frac{\vec{F}_{n+1,\cdot}^{\mathrm{T}}(I_n-F_n)^{-1}}{\Delta_{n+1}} & \frac{1}{\Delta_{n+1}} \end{bmatrix} \qquad (9)
\end{aligned}
$$

According to Corollary 1,

$$
\begin{aligned}
\begin{bmatrix} P_n^{*} \\ p_{n+1}^{*} \end{bmatrix} &= (I_{n+1} - F_{n+1})^{-1} \begin{bmatrix} U_n \\ u_{n+1} \end{bmatrix} \\
&= \begin{bmatrix} (I_n - F_n)^{-1}U_n + \frac{(I_n-F_n)^{-1}\vec{F}_{\cdot,n+1}\vec{F}_{n+1,\cdot}^{\mathrm{T}}(I_n-F_n)^{-1}}{\Delta_{n+1}}U_n + \frac{(I_n-F_n)^{-1}\vec{F}_{\cdot,n+1}}{\Delta_{n+1}}u_{n+1} \\ \frac{\vec{F}_{n+1,\cdot}^{\mathrm{T}}(I_n-F_n)^{-1}U_n + u_{n+1}}{\Delta_{n+1}} \end{bmatrix} \\
&= \begin{bmatrix} P_{n,old}^{*} + (I_n - F_n)^{-1}\vec{F}_{\cdot,n+1}p_{n+1}^{*} \\ \frac{\gamma_{n+1}}{G_{n+1,n+1}\Delta_{n+1}}R_{n+1}^{\text{initial}} \end{bmatrix}. \qquad (10)
\end{aligned}
$$

Further, by (6), mobile m_{n+1} is admissible if and only if $p_{n+1}^{*} > 0$ and $P_n^{*} > \vec{0}_n$, which is equivalent to $\Delta_{n+1} > 0$, since the n existing downlinks are feasible and the matrix $(I_n - F_n)^{-1}$ is componentwise positive. □

By Theorem 2, the inequality $P_n^{*} > P_{n,old}^{*}$ always holds. In other words, every time a new call is admitted, all the existing downlinks have to increase their powers to overcome the interference resulting from the new call (and this is true even if the call is not yet admitted into the system). The quantity $\vec{F}_{\cdot,n+1}p_{n+1}^{*}$ in (8) is the extra power required to overcome the interference from the downlink of m_{n+1} alone. The matrix $(I_n - F_n)^{-1}$ represents the coupling between the n existing downlinks.

The quantity $\frac{\gamma_{n+1}R_{n+1}^{\text{initial}}}{G_{n+1,n+1}} \overset{\text{def}}{=} p_{n+1}^{\text{initial}}$ is the transmitted power needed by base station B_{n+1} to overcome R_{n+1}^{initial}, which is the total interference power received by mobile m_{n+1} when it arrives. However, there is a self-coupling effect for the new mobile m_{n+1}: the initial power transmitted at the downlink of m_{n+1} results in a power increase of the n existing downlinks, and this increase will in turn make the downlink transmitted power of m_{n+1} itself increase. This effect is totally captured by the discriminant Δ_{n+1}, because

$$p_{n+1}^{*} = p_{n+1}^{\text{initial}}/\Delta_{n+1}. \qquad (11)$$

Note that the discriminant Δ_{n+1} only depends on information contained in matrix F_{n+1}, so it is a parameter inherent to the system. Unfortunately, the matrix F_{n+1} involves global information. In fact, the quantity $1/\Delta_{n+1}$ is just the $(n+1)^{st}$ pivoting variable in the Gaussian reduction tableau, which results in a centralized CAC criterion [2]. Therefore, the discriminant by itself appears to be unsuitable as a distributed CAC criterion. We next provide a technique to overcome this limitation, resulting in a distributed CAC scheme.

3.2 Distributed CAC Schemes

A diagram illustrating the centralized power assignment scheme (8) is shown in Figure 1 (the thick lines represent vector information flows, and the thin lines represent scalar

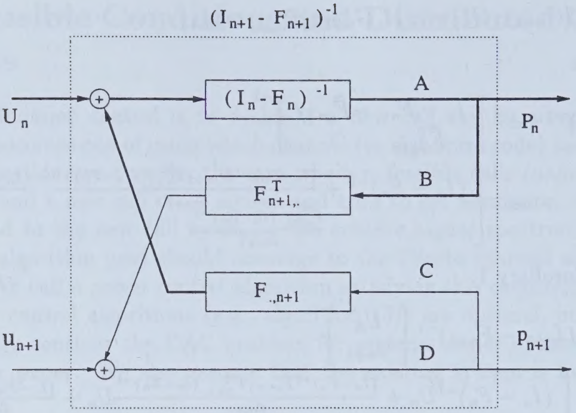


Figure 1: Power assignment for $n+1$ users.

flows). For m_{n+1}, there is a feedback loop (with gain $1 - \Delta_{n+1}$), which gives rise to the "self-coupling" effect and the problem of feasibility.

Before admitting mobile m_{n+1}, the n existing downlinks are feasible. If we cut the forward path D in Figure 1, i.e., let base station B_{n+1} communicate with m_{n+1} at a fixed power level (we take p^{initial}_{n+1} defined above, to keep the initial SIR of the downlink at γ_{n+1}), then the feasibility of the n existing downlinks will not change. This is true because, for the existing downlinks, the additional interference resulting from m_{n+1} is equivalent to an increase in U_n. The existing downlinks only have to increase the transmitted power from $P^*_{n,old}$ to $P^*_{n,old} + (I_n - F_n)^{-1}\vec{F}_{\cdot,n+1}p^{\text{initial}}_{n+1}$, to maintain their desired SIR's. After this increase, the downlink of m_{n+1} has SIR lower than γ_{n+1}, because the received interference changes to

$$R^{\text{new}}_{n+1} \triangleq R^{\text{initial}}_{n+1} + [G_{n+1,1}, G_{n+1,2}, \ldots, G_{n+1,n}](I_n - F_n)^{-1}\vec{F}_{\cdot,n+1}p^{\text{initial}}_{n+1} > R^{\text{initial}}_{n+1}. \qquad (12)$$

To increase the SIR back to γ_{n+1}, base station B_{n+1} has to increase the transmitted power for m_{n+1} by $\vec{F}^{T}_{n+1,\cdot}(I_n - F_n)^{-1}\vec{F}_{\cdot,n+1}p^{\text{initial}}_{n+1}$. We define:

$$\begin{aligned} p^{\text{new}}_{n+1} &= \frac{\gamma_{n+1}R^{\text{new}}_{n+1}}{G_{n+1,n+1}} && (13) \\ &= p^{\text{initial}}_{n+1} + \vec{F}^{T}_{n+1,\cdot}(I_n - F_n)^{-1}\vec{F}_{\cdot,n+1}p^{\text{initial}}_{n+1} \\ &= p^{\text{initial}}_{n+1} + (1 - \Delta_{n+1})p^{\text{initial}}_{n+1}. \end{aligned}$$

Then, the above equation yields

$$\Delta_{n+1} = 2 - \frac{p^{\text{new}}_{n+1}}{p^{\text{initial}}_{n+1}} = 2 - \frac{R^{\text{new}}_{n+1}}{R^{\text{initial}}_{n+1}}. \qquad (14)$$

By Theorem 2, we have

Corollary 3 *Mobile m_{n+1} is admissible if and only if $p^{\text{new}}_{n+1} < 2p^{\text{initial}}_{n+1}$, which is equivalent to $R^{\text{new}}_{n+1} < 2R^{\text{initial}}_{n+1}$.*

Note that quantities p_{n+1}^{initial}, p_{n+1}^{new}, R_{n+1}^{initial}, and R_{n+1}^{new} are independent of the iterative power control algorithm used. The CAC criterion only requires the algorithm to be distributed and optimal. Thus, the admissibility criterion given above is very general. Most important, we can check the criterion in a distributed and asynchronous manner as described next.

In the above derivation, we assume that the downlink power of m_{n+1} is fixed at p_{n+1}^{initial}. However, as mentioned before, arbitrary positive $p_{n+1}(0)$ in place of p_{n+1}^{initial} works. Then we have

$$\Delta_{n+1} = 1 - \frac{\gamma_{n+1}(R'_{n+1} - R_{n+1}^{\text{initial}})}{G_{n+1,n+1}p_{n+1}(0)}, \tag{15}$$

where $R'_{n+1} = R_{n+1}^{\text{initial}} + [G_{n+1,1}, G_{n+1,2}, \ldots, G_{n+1,n}](I_n - F_n)^{-1}\vec{F}_{\cdot,n+1}p_{n+1}(0)$ is analogous to R_{n+1}^{new}. With the distributed computation of Δ_{n+1}, we have the following distributed CAC algorithm:
Δ-CAC

1. Mobile m_{n+1} measures and reports to base station B_{n+1} its received interference power R_{n+1}^{initial}. The base station transmits to m_{n+1} at a fixed power level $p_{n+1}(0)$.
2. The downlinks of the n exisiting mobiles increase their corresponding powers iteratively according to some distributed, optimal power control algorithm (e.g., (3)).
3. After convergence, mobile m_{n+1} reports its received interference power R'_{n+1} to base station B_{n+1}. The base station calculates Δ_{n+1} according to (15). If $\Delta_{n+1} \leq 0$, then the new call is rejected. Otherwise, the base station admits the call, and the $(n+1)$ downlinks adjust the transmitted power iteratively to the Pareto optimal assignment.

According to Theorem 2, if the system becomes infeasible, mobile m_{n+1} is rejected by the algorithm; otherwise it is admitted, and the system will converge to the optimal power assignment. Note that Δ-CAC makes the admission decision after the system converges, so the delay until a call is rejected may be large, and in a practical setting it may not be easy to deal with a new arrival or a departure that occurs during the admission procedure. We next present another algorithm to overcome this limitation.

If an optimal power control algorithm monotonically increases the transmitted powers (from the steady state values) during the admission process, we call it *monotonic and optimal.* An example of such an algorithm is algorithm (3). It should be pointed out, however, that the SIR's may not increase monotonically during the admission process. The monotonic and optimal power control is of special interest and importance in CAC, because it is less aggressive and will result in a more stable system. Note that the interference R_{n+1} received by mobile m_{n+1} will increase monotonically during the iterations of a monotonic and optimal power control. In this case, we can use the result of Corollary 3 as a CAC criterion, which is better than $\Delta_{n+1} > 0$, because using it we do not need to wait until convergence to reject an infeasible call; thus resulting in a smaller disturbance. Based on the new criterion, we have a simplified and improved distributed admission algorithm:
R-CAC

1. Mobile m_{n+1} measures its total interference R_{n+1}^{initial} and reports this value to its home base station B_{n+1}. The base station transmits to mobile m_{n+1} at fixed power level $p_{n+1}^{\text{initial}} = \frac{\gamma_{n+1}}{G_{n+1,n+1}} R_{n+1}^{\text{initial}}$.

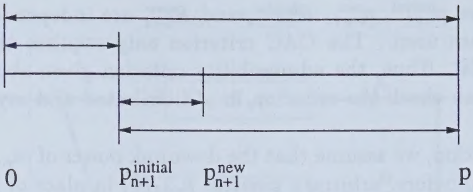


Figure 2: $p_{n+1}^{\text{initial}}/p_{n+1}^{*} = (p_{n+1}^{\text{new}} - p_{n+1}^{\text{initial}})/(p_{n+1}^{*} - p_{n+1}^{\text{initial}}) = \Delta_{n+1}$.

2. The downlinks of the n exisiting mobiles increase their corresponding powers iteratively according to some distributed, monotonic and optimal power control algorithm.

3. During the process of iteration mobile m_{n+1} measures its received interference power R_{n+1}. If it reaches (or exceeds) $2R_{n+1}^{\text{initial}}$, then the new call is rejected. Otherwise, after convergence, it is admitted. Mobile m_{n+1} measures and reports its received interference power R_{n+1}^{new} to base station B_{n+1}. The base station calculates Δ_{n+1}, and adjusts the transmitted power to $\frac{p_{n+1}^{\text{initial}}}{\Delta_{n+1}}$; the n existing downlinks correspondingly increase their associated power levels.

3.3 Further Discussions on the Discriminant

Having discussed the CAC algorithms, we now further elaborate on the discriminant Δ_{n+1}, which plays a fundamental role in this paper. From Theorem 2, if mobile m_{n+1} is admissible, then we have $\Delta_{n+1} = \frac{p_{n+1}^{\text{initial}}}{p_{n+1}^{*}}$ by (11). Combining this with Equation (14) yields

$$\Delta_{n+1} = \frac{p_{n+1}^{\text{initial}}}{p_{n+1}^{*}} = \frac{p_{n+1}^{\text{new}} - p_{n+1}^{\text{initial}}}{p_{n+1}^{*} - p_{n+1}^{\text{initial}}}. \tag{16}$$

We illustrate the above relationship in Figure 2. Clearly, the condition $p_{n+1}^{\text{new}} = 2p_{n+1}^{\text{initial}}$ will drive p_{n+1}^{*} to ∞ (i.e., system diverges), because $p_{n+1}^{\text{initial}} > 0$. This provide further insight into our CAC criterion.

By fixing p_{n+1}^{initial} at 1, we obtain $p_{n+1}^{\text{new}} = 2 - \Delta_{n+1}$; and if m_{n+1} is admissible, then $p_{n+1}^{*} = 1/\Delta_{n+1}$. Figure 3 shows the three powers p_{n+1}^{initial}, p_{n+1}^{new}, and p_{n+1}^{*}, versus Δ_{n+1}. The plot ends at $\Delta_{n+1} = 1$ because the n existing calls are feasible, and so by definition $\Delta_{n+1} \leq 1$. The three powers are equal at $\Delta_{n+1} = 1$, which corresponds to $n = 0$. When m_{n+1} is not admissible (i.e., $\Delta_{n+1} \leq 0$), p_{n+1}^{*} goes to ∞, but p_{n+1}^{new} remains bounded and small. This observation illustrates the benefit of using our CAC criterion together with a power control algorithm compared to using the power control algorithm by itself. To elaborate, we start from p_{n+1}^{initial} to get p_{n+1}^{new}, then decide on admissibility, and only when m_{n+1} is admissible will the power be tuned to p_{n+1}^{*}. In contrast, in the case of using power control by itself, the scheme tries to achieve p_{n+1}^{*} directly, and may fail, resulting in divergence. Note that though fixed to 1 here, the power p_{n+1}^{initial} depends on Δ_{n+1}. Qualitatively, a smaller Δ_{n+1} corresponds to a larger interference R_{n+1}^{initial}, and hence also a larger p_{n+1}^{initial}, which worsens the situation when m_{n+1} is not admissible.

The monotonicity of p_{n+1}^{*} with Δ_{n+1} also suggests that a bigger Δ_{n+1} indicates a more robust system. To prevent the transmitted powers and admission delays becoming

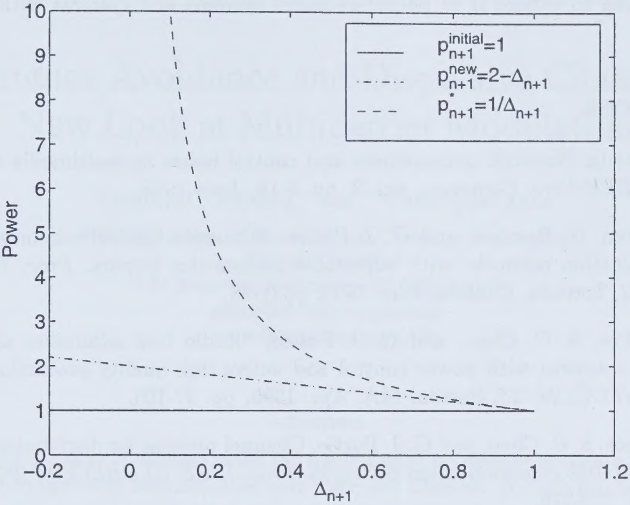


Figure 3: Powers versus Δ_{n+1}.

impractically large, we need to leave some robustness margin for the system during the admission process. This can be achieved by modifying the above CAC criteria into $\Delta_{n+1} > \varepsilon$, where ε is a predefined small positive number.

By estimating the discriminant on-line, we can also solve the problems of channel selection, handoff prioritization, base station assignment, and wideband traffic. Due to the length limit of the paper, we have to refer readers to [12] for details on these topics, and for numerical results as well.

4 Conclusion and Further Work

We have studied the connection admission control problem for cellular systems under optimal distributed power control. These systems can achieve high spectrum utilization; however, if an infeasible call is admitted, the power levels may diverge and result in dropping an existing call.

We defined for the newly arrived mobile a system parameter, called the discriminant, which provides an admissibility criterion. Based on this criterion, we proposed a distributed CAC scheme called Δ-CAC. One disadvantage of Δ-CAC is that it has to wait until the system settles down to make an admission decision. We overcame this limitation, for systems under monotonic and optimal power control, by using a simpler distributed scheme called R-CAC. Both algorithms are distributed, in the sense that they only require locally available information. Our CAC schemes for power-controlled systems are general, because we need not assume any particular multiple access technique, specific power control algorithm, or knowledge about the traffic. In combination with DCA and power control, our CAC schemes can greatly improve the spectrum efficiency over traditional CAC schemes.

This work only considers circuit-switched cellular systems without power constraints.

We are working to extend it to packet-switched systems and systems with power constraints.

References

[1] M. Schwartz. Network management and control issues in multimedia wireless networks. *IEEE Pers. Commun.*, vol. 2, pp. 8-16, June 1995.

[2] S. C. Chen, N. Bambos, and G. J. Pottie. Admission control schemes for wireless communication networks with adjustable transmitter powers. *Proc. IEEE INFOCOM '94*, Toronto, Canada, June 1994, pp21-28.

[3] N. Bambos, S. C. Chen, and G. J. Pottie, "Radio link admission algorithm for wireless networks with power control and active link quality protection," in *Proc. IEEE INFOCOM '95*, Boston, MA, Apr. 1995, pp. 97-104.

[4] N. Bambos, S. C. Chen, and G. J. Pottie. Channel probing for distributed access control in wireless communication networks. *Proc. IEEE GLOBECOM '95*, Singapore, 1995, pp. 322-326.

[5] N. Bambos. Toward power-senstive network architectures in wireless communications: concepts, issues, and design aspects. *IEEE Pers. Commun.*, vol. 5, pp. 50-59, June, 1998.

[6] D. Mitra. An asynchronous distributed algorithm for power control in cellular radio systems. *Proc. 4th Winlab Workshop Third Generation Wireless Infomation Network*, Rutgers Univ., Oct. 1993.

[7] R. D. Yates. A framework for uplink power control in cellular radio systems. *IEEE J. Select. Areas Commun.*, vol.13, pp. 1341-1347, Sept. 1995.

[8] S. V. Hanly. Congestion measures in DS-CDMA networks. *IEEE Trans. Comm.*, to appear.

[9] E. Seneta. *Non-negative matrices*, John Wiley & Sons, New York, 1973.

[10] C.-Y. Huang. Radio resource management in power controlled CDMA systems. *PhD thesis*, Rutgers University, 1996.

[11] J. Zander. Performance of optimum transmitter power control in cellular radio systems. *IEEE Trans. Veh. Technol.*, vol. 41, pp. 57-62, Feb. 1992.

[12] M. Xiao, N. B. Shroff, and E. K. P. Chong. Distributed admission control for power-controlled cellular wireless systems, preprint, Sept. 1999.

Interference Avoidance and Dispersive Channels: A New Look at Multicarrier Modulation

Dimitrie C. Popescu and Christopher Rose
WINLAB, Dept. of Electrical and Computer Engineering
Rutgers University
94 Brett Rd., Piscataway, NJ 08854-8058
cripop@winlab.rutgers.edu
crose@winlab.rutgers.edu

Abstract

The availability of software radios and application of interference avoidance may offer a new perspective on communication over dispersive channels. By both lengthening the transmission interval so that ISI is almost inconsequential and by using combinations of channel eigenfunctions to convey the bits, communications over dispersive channels becomes (not surprisingly) a multiuser detection problem where each bit is assigned its own waveform. By choosing these waveforms to meet the Welch Bound with equality – through interference avoidance or some other numerical method – a uniform maximum SINR can be achieved for each bit. In addition by using Guassian signaling and decoding with this codeword ensemble, the channel capacity can be (theoretically) met. A trivial extension of the single user case can be used for multiple users over identical channels. Multiple users with non-identical channels is more subtle and is discussed in the interference avoidance context with suggestions for future work.

1 Introduction

Wireless communication channels are generally dispersive and dispersion leads to intersymbol interference (ISI) where energy from signal bits spills over into the observation intervals of subsequent bits at the receiver. As noted in [1] there have been primarily two ways to combat ISI – equalization and coding – with large associated literatures.

In [1] and [2] an approach was proposed based on partitioning the communication channel into a set of parallel and independent subchannels each associated with its own carrier. Such channel partitioning is known in general as multicarrier modulation and has been present in the literature for over 30 years. The specific approach employed by Kasturia *et al* was dubbed vector coding.

Our approach is also based on a form of multicarrier modulation that uses the eigenfunctions of the channel autocorrelation [3,4]. Similar to [1,2], vectors are sent over the channel during some communications interval of duration $\mathcal{T}$ where $\mathcal{T}$ is assumed long enough so that the duration of the channel impulse response is small in comparison. However, our approach differs in that we assign different linear combinations of eigenfunctions to *each bit* sent during the $\mathcal{T}$ communications interval. That is, the signature for *each bit* is assumed to be of duration $\mathcal{T}$ – similar to, but different from prior work where bit signatures were allowed to have length longer than the bit interval [5].

Furthermore, the signatures are chosen to meet the Welch Bound with equality (WBE sequences) so that the *optimal* linear receiver is a bank of matched filters tuned to channel response to each signature [6,7]. Such WBE sequences achieve the maximum attainable uniform signal to interference/noise ratio (SINR) per bit, and if used with Gaussian signaling could be (theoretically) used to meet the capacity of the channel.[1]

From another perhaps simpler perspective, by using channel eigenfunctions as modulation building blocks we transform the usual dispersive channel problem into a multiuser detection problem. Although this isomorphism is not unexpected, it proves useful in the interference avoidance context. Specifically, we start by casting the problem of sending a set of M independent bits over a channel during an interval of duration $\mathcal{T}$ as a problem in multiuser detection by using linear combinations of channel eigenfunctions of the channel autocorrelation (codewords) for transmission of each bit. Then we apply interference avoidance techniques [8–10] to determine optimal codewords for each bit. Since it can be shown that greedy interference avoidance always produces codeword ensembles which meet the Welch bound with equality [11], the resulting codeword sets maximize ensemble SINR and can also be used (theoretically with Gaussian signaling and decoding) to maximize channel capacity.

The simplest multiple user problem, where L users share an identical channel for communication is a simple extension of the single user problem. In this case, the "user capacity" of the channel is maximized in that the scheme supports the greatest number of users subject to a given SINR criterion. And as before, under a Gaussian signaling assumption, the sum capacity is achieved as well. The multiple user with non-identical channels case is more subtle and will be discussed.

2 Background: channel eigenfunctions

Suppose a channel can be characterized by a causal impulse response $h(t)$. We will now (re)derive [3,4] a useful set of functions which will allow us to easily represent channel inputs and outputs. Specifically we desire a set of orthonormal basis functions $\{\Psi_i(t)\}$ which can be used to represent channel inputs. However, in addition, we would like the channel responses to these functions to be orthogonal as well to allow simple representation of outputs. That is, on the communications interval $(0, \mathcal{T})$ where $\mathcal{T}$ is assumed much larger than the duration of $h(t)$ we require

$$\int_0^{\mathcal{T}} \left[\int_0^{\mathcal{T}} \Psi_i(\tau)h(t-\tau)d\tau\right]\left[\int_0^{\mathcal{T}} \Psi_j(\mu)h(t-\mu)d\mu\right] dt = \lambda_i\delta_{ij} \tag{1}$$

Defining a channel correlation function as

$$R_h(\tau-\mu) = \int_0^{\mathcal{T}} h(t-\tau)h(t-\mu)dt \tag{2}$$

we obtain

$$\int_0^{\mathcal{T}}\int_0^{\mathcal{T}} \Psi_j(\mu)\Psi_i(\tau)R_h(\tau-\mu)d\tau d\mu = \lambda_i\delta_{ij} \tag{3}$$

Following [12], we readily see that the desired $\{\Psi_i(t)\}$ must satisfy the integral equation

$$\int_0^{\mathcal{T}} \Psi_i(\mu)R_h(\tau-\mu)d\mu = \lambda_i\Psi_i(\tau) \tag{4}$$

---

[1] Assuming the channel impulse response duration is negligible compared to the communications interval duration $\mathcal{T}$.

and furthermore since $R_h()$ is a correlation function, a complete set of orthonormal $\Psi_i(\tau)$ exists along with a corresponding set of non-negative λ_i. These $\{\Psi_i(t)\}$ *eigenfunctions* can therefore be used to represent any input in the mean square sense. Similarly, any feasible channel output can be written as a linear superposition of the set $\{\tilde{\Psi}_i(t)\}$ where we define

$$\tilde{\Psi}_i(t) = \lambda_i^{-\frac{1}{2}} \int_0^{\mathcal{T}} \Psi_i(\tau) h(t-\tau) d\tau \tag{5}$$

for non-zero λ_i. It is worth noting that each of the $\Psi_i(t)$ and $\tilde{\Psi}_i(t)$ is assumed to have unit energy ($\int_0^{\mathcal{T}} \Psi_i^2(t)dt = \int_0^{\mathcal{T}} \tilde{\Psi}_i^2(t)dt = 1$). So if we have as input to the channel

$$x(t) = \sum_i s_i \Psi_i(t) \tag{6}$$

the corresponding channel output is then simply

$$y(t) = \sum_i s_i \lambda_i^{\frac{1}{2}} \tilde{\Psi}_i(t) \tag{7}$$

Now suppose that there are only N non-negligible eigenvalues λ_i. This implies that the useful input space – that portion of the signal space whose component energies are not strongly absorbed by the channel – is finite. We note that such a condition is unavoidable for finite $\mathcal{T}$ and finite bandwidth constraints as would be introduced by any real $h(t)$. We can then closely approximate useful inputs as

$$x(t) = \mathbf{P}(t)^{\top} \mathbf{s} \tag{8}$$

where $\mathbf{s}$ is a column vector of N s_i's and $\mathbf{P}(t)$ is a column vector of the significant N $\Psi_i(t)$'s. Likewise we have as channel output

$$y(t) = \tilde{\mathbf{P}}(t)^{\top} \Lambda^{1/2} \mathbf{s} \tag{9}$$

for outputs where Λ is the $N \times N$ diagonal matrix of the significant eigenvalues.

3 Multicarrier Modulation

The usual bit-sequential scenario would have

$$x(t) = \sum_k b_k p(t-kT) \tag{10}$$

where b_k is the bit sent during interval k, T is the bit interval and $p(t)$ is the finite duration pulse used to convey the information over the channel. This signaling structure is depicted in FIGURE 1.

Multicarrier modulation offers a different approach. Let $\mathcal{T}$ be some interval of duration much longer than the channel impulse response duration. In this way we can safely ignore ISI between successive intervals of duration $\mathcal{T}$ by placing a relatively small "zero pad" between them to allow settling of the channel response. We will send M bits during the interval and construct the input waveform as

$$x(t) = \sum_{k=1}^{M} b_k s_k(t) \tag{11}$$

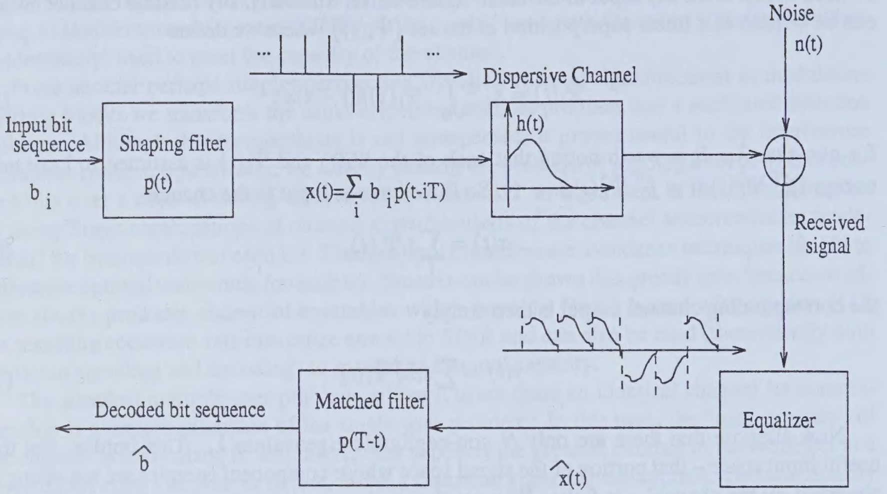

Noise
n(t)
...
...
Dispersive Channel
Input bit sequence
Shaping filter
p(t)
h(t)
b_i
$x(t)=\sum_i b_i p(t-iT)$
Received signal
Matched filter
p(T-t)
Decoded bit sequence
$\hat{b}$
Equalizer
$\hat{x}(t)$

Figure 1: A depiction of sequential transmission with equalization.

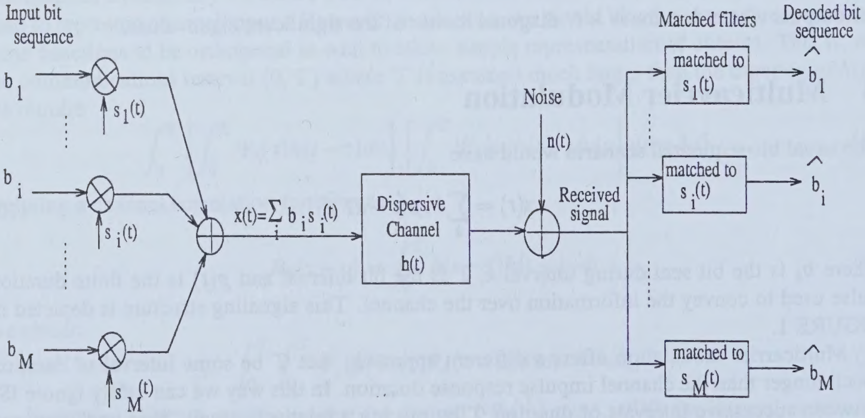

Input bit sequence
b_1
$s_1(t)$
b_i
$s_i(t)$
b_M
$s_M(t)$
$x(t)=\sum_i b_i s_i(t)$
Dispersive Channel
h(t)
Noise
n(t)
Received signal
Matched filters
matched to $s_1(t)$
matched to $s_i(t)$
matched to $s_M(t)$
Decoded bit sequence
$\hat{b}_1$
$\hat{b}_i$
$\hat{b}_M$

Figure 2: A depiction of our version of multicarrier modulation.

where $s_k(t)$ is a waveform chosen specifically to convey bit i. Unlike [1,2] there is no explicit precoding. Rather, the method would be a form of CDMA as if each bit corresponded to a different user as shown in FIGURE 2.

Each of the waveforms $s_k(t)$ can be represented in terms of the spanning set of previously described orthonormal basis functions $\{\Psi_j(t)\}_{j=1}^N$ as

$$s_k(t) = \sum_{j=1}^{N} s_{kj}\Psi_j(t) \tag{12}$$

which implies that the input signal can be written as

$$x(t) = \sum_{k=1}^{M} b_i \sum_{j=1}^{N} s_{kj}\Psi_j(t) = \mathbf{P}(t)^T\mathbf{S}\mathbf{b} \tag{13}$$

where $\mathbf{b} = [b_1 \dots b_M]^T$ is the vector containing the bit set to be sent, $\mathbf{P}(t) = [\Psi_1(t) \dots \Psi_N(t)]^T$ is a vector containing the basis functions, and $\mathbf{S}$ an $N \times M$ matrix with columns $\mathbf{s}_k$ being the coordinate vectors of each waveform $\{s_k(t)\}_{k=1}^M$ with respect to the basis $\{\Psi_j(t)\}_{j=1}^N$.

The received signal is assumed to contain additive zero mean white Gaussian noise and can be represented as

$$r(t) = y(t) + n(t) \tag{14}$$

which when projected onto the orthogonal functions $\{\tilde{\Psi}_i(t)\}_{i=1}^N$ becomes

$$\mathbf{r} = \Lambda^{1/2}\mathbf{S}\mathbf{b} + \mathbf{n} \tag{15}$$

with $\mathbf{n} = [n_1 \dots n_N]^T$ a vector containing the projections of the noise process $n(t)$ onto the basis functions – $n_j = \int_0^T n(t)\tilde{\Psi}_j(t)dt$. Since $n(t)$ is a white Gaussian process, the projections onto orthogonal functions are mutually independent zero mean Gaussian random variables with variances $E[n_j^2] = N_0$.

Since we have limited the input signal space to those $\Psi_i(t)$ with nonzero eigenvalues, we can rewrite equation (15) to obtain

$$\tilde{\mathbf{r}} = \Lambda^{-1/2}\mathbf{r} = \mathbf{S}\mathbf{b} + \tilde{\mathbf{n}} \tag{16}$$

in which $\tilde{\mathbf{n}} = \Lambda^{-1/2}\mathbf{n}$ is a new vector of noise components, still uncorrelated, but no longer white – the variance of each component is now N_0/λ_i. This is a typical multiuser detection problem [13] where an ensemble of user transmissions $b_k\mathbf{s}_k$ must be jointly decoded subject to a constraint on transmitted energy for each codeword. Thus, as mentioned previously the optimal multicarrier modulation problem is (not surprisingly) isomorphic with the optimal multiuser detection problem posed in equation (16).

At this point we could turn to multiuser detection theory and seek to design the best receivers (decision feedback, MMSE etc.) for the given signatures. However, we instead pursue an alternate approach of signature optimization. This will allow us to use simple matched filters for optimal detection in the linear receiver case.

4 Interference Avoidance

We now require a means to generate such optimal signal sets. To this end a class of *interference avoidance* algorithms has been proposed [8–10] in which the signature waveform of a given user is adapted to minimize interference from other users and achieve maximum signal-to-interference ratio. This procedure in general has been shown empirically to produce optimal signal sets and has theoretical properties which suggest convergence is unavoidable [8]. Recently however [11], we have found a strict convergence proof for a subclass of interference avoidance algorithm dubbed "greedy interference avoidance." Thus, we can be assured that interference avoidance will yield an optimal signal set.

We will first introduce the greedy *eigen-algorithm* [9,10] in the current context and describe properties of the optimal signal set. Note that to simplify the argument and avoid certain special cases we hereafter assume that the number of bits to be sent M is at least as great as the number of eigenfunctions N.

Greedy interference avoidance for minimizing interference from other bit transmissions can be summarized as follows:

- Start with an initial set of signatures $\{s_k(t)\}_{k=1}^{M}$, with corresponding signature vectors $\{\mathbf{s}_k\}_{k=1}^{M}$ with respect to the orthonormal basis defined by the channel eigenfunctions.
- For each bit k, the interference generated by the other $M-1$ independent bits is described by the N-dimensional autocorrelation matrix $\mathbf{R}_k = \mathbf{S}\mathbf{S}^\top - \mathbf{s}_k\mathbf{s}_k^\top + N_0\Lambda^{-1}$.
- The minimum eigenvalue μ_k^* of $\mathbf{R}_k$ and its associated eigenvector $\mathbf{x}_k^*$ are found, and then $\mathbf{s}_k$ is replaced by $\mathbf{x}_k^*$.
- This process is repeated iteratively for each bit signature until no further improvement can be obtained. At that point, all the codewords are minimum eigenvalue eigenvectors of $\mathbf{R}_k$.
- If a suboptimal minimum is reached, employ "class warfare" methods to escape the local minimum. A finite number of minima guarantees convergence [11].

It can be shown that this procedure monotonically decreases an intuitively pleasing quantity called the total square correlation (TSC) [8–10], defined as

$$\mathrm{TSC} = \mathrm{Trace}[(\mathbf{S}\mathbf{S}^\top + N_0\Lambda^{-1})^2] \tag{17}$$

Furthermore, it can be shown by simple extension of majorization [14] results in [7] or by direct methods using Lagrange multipliers and the Kuhn-Tucker condition [11] that minimization of TSC results in a codeword set which maximizes sum capacity and user capacity. Remember that user capacity is defined as the maximum number of users (codewords) which meet a specified SINR [6,7].

We now seek the minimum TSC. Let us write the matrix $\mathbf{S}\mathbf{S}^\top + N_0\Lambda^{-1}$ in terms of canonical eigenvectors $\mathbf{e}_i = [0,\ldots,0,1,0,\ldots,0]^\top$ (zero except for 1 in the i^{th} position) of $\Re^N$. Then we have

$$\begin{aligned} \mathbf{S}\mathbf{S}^\top + N_0\Lambda^{-1} &= \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{k=1}^{M} s_{ik}s_{jk}\mathbf{e}_i\mathbf{e}_j^\top + \sum_{i=1}^{N}\sum_{j=1}^{N}\frac{N_0}{\lambda_j}\mathbf{e}_i\mathbf{e}_j^\top \\ &= \sum_{i=1}^{N}\sum_{j=1}^{N}\left(\sum_{k=1}^{M} s_{ik}s_{jk} + \frac{N_0}{\lambda_j}\right)\mathbf{e}_i\mathbf{e}_j^\top \end{aligned} \tag{18}$$

We can then write

$$
\begin{aligned}
(\mathbf{S}\mathbf{S}^\top + N_0\Lambda^{-1})^2 &= \left[\sum_{i=1}^{N}\sum_{j=1}^{N}\left(\sum_{k=1}^{M} s_{ik}s_{jk} + \frac{N_0}{\lambda_j}\right)\mathbf{e}_i\mathbf{e}_j^\top\right] \\
&\quad \times \left[\sum_{m=1}^{N}\sum_{n=1}^{N}\left(\sum_{l=1}^{M} s_{ml}s_{nl} + \frac{N_0}{\lambda_l}\right)\mathbf{e}_m\mathbf{e}_n^\top\right] \\
&= \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{n=1}^{N}\sum_{k=1}^{M}\left(s_{ik}s_{jk} + \frac{N_0}{\lambda_j}\right)\sum_{l=1}^{M}\left(s_{jl}s_{nl} + \frac{N_0}{\lambda_j}\right)\mathbf{e}_i\mathbf{e}_n^\top
\end{aligned}
\tag{19}
$$

Since $\mathrm{Trace}[\mathbf{e}_i\mathbf{e}_n^\top] = \mathbf{e}_n^\top\mathbf{e}_i = \delta_{in}$, we then have

$$
\mathrm{Trace}[(\mathbf{S}\mathbf{S}^\top + N_0\Lambda^{-1})^2] = \sum_{i=1}^{N}\sum_{j=1}^{N}\left(\sum_{k=1}^{M} s_{ik}s_{jk} + \frac{N_0}{\lambda_j}\right)^2 \tag{20}
$$

By defining the total signal energy projection along the i^{th} eigenfunction as $P_i = \sum_{k=1}^{M} s_{ik}^2$ we can rewrite (20) as

$$
\mathrm{Trace}[(\mathbf{S}\mathbf{S}^\top + N_0\Lambda^{-1})^2] = \sum_{i=1, i\neq j}^{N}\sum_{j=1}^{N}\left(\sum_{k=1}^{M} s_{ik}s_{jk} + \frac{N_0}{\lambda_j}\right)^2 + \sum_{j=1}^{N}\left(P_j + \frac{N_0}{\lambda_j}\right)^2 \tag{21}
$$

Since all terms in the summations in i and j are non-negative we can lower bound the TSC by

$$
\mathrm{Trace}[(\mathbf{S}\mathbf{S}^\top + N_0\Lambda^{-1})^2] \geq \sum_{j=1}^{N}\left(P_j + \frac{N_0}{\lambda_j}\right)^2 \tag{22}
$$

where the right hand side term is a convex function in P_j. Minimization of this term with the requirements that $P_j \geq 0$, $\forall j$ and $\sum_{j=1}^{N} P_j = M$ (total power sent – M bits each with unit energy) leads to the "water filling" distribution of powers [3,6,7,15]

$$
P_i = \left(c^* - \frac{N_0}{\lambda_i}\right)^+, \quad \text{where } (x)^+ = \begin{cases} x & \text{if } x \geq 0 \\ 0 & \text{if } x < 0 \end{cases} \tag{23}
$$

where c^* is the water level mark and can be calculated from the total power constraint. It will later be useful to note that the above derivation also applies to correlated noise. That is, for a non-diagonal noise covariance matrix, the noise covariance diagonalizing similarity transform could first be applied to obtain the uncorrelated noise form similar to equation (19).

We then remember [16] that if the eigenvalues of a matrix $\mathbf{A}$ are μ_i then the eigenvalues of $\mathbf{A}^2$ are μ_i^2. Likewise we remember that $\mathrm{Trace}[\mathbf{A}] = \sum_i \mu_i$. Thus,

$$
\mathrm{Trace}[(\mathbf{S}\mathbf{S}^\top + N_0\Lambda^{-1})^2] = \sum_{i=1}^{N} \beta_i^2 \tag{24}
$$

where the $\{\beta_i\}$ are the eigenvalues of $\mathbf{S}\mathbf{S}^\top + N_0\Lambda^{-1}$. If these β_i are chosen to be the $P_i + N_0/\lambda_i$ where the P_i satisfy equation (23), then the bound of equation (22) is met with equality. Finally, the existence of such codeword sets $\{\mathbf{s}_k\}$ is guaranteed by the provable convergence to the minimum TSC of greedy interference avoidance with class warfare [11].

So in summary, once the single-user intersymbol interference problem is recast as a multiuser detection problem, codeword adaptation can be brought to bear and an optimal codeword found for each bit to be transmitted by the user. Assuming simple matched filters for each bit's codeword, the bits each attain the maximum achievable uniform SINR. Furthermore, from a user capacity perspective, simple matched filter receivers *are optimal* [6,7].

5 Extension to Multiple Users

5.1 Identical Channels

An extension to multiple users sharing the same dispersive channel is trivial. Letting each of the L users have their own codeword ensemble matrix $\mathbf{S}_i$ with M codeword columns $\mathbf{s}_{ik}$ $i = 1,2,\cdots,M$ and bit vectors $\mathbf{b}_i$ of dimension M, the received signal is

$$\mathbf{r} = \mathbf{n} + \Lambda^{1/2} \sum_{i=1}^{L} \mathbf{S}_i \mathbf{b}_i = \mathbf{n} + \Lambda^{1/2} \mathbf{S}' \mathbf{b}' \tag{25}$$

where

$$\mathbf{S}' = [\mathbf{s}_{11} \cdots \mathbf{s}_{1M} | \mathbf{s}_{21} \cdots \cdots \mathbf{s}_{LM}] \tag{26}$$

and

$$\mathbf{b}' = \begin{bmatrix} \mathbf{b}_1 \\ \mathbf{b}_2 \\ \vdots \\ \mathbf{b}_L \end{bmatrix} \tag{27}$$

Equation (25) is identical in form to equation (15) so the same single user results pertain.

5.2 Non-Identical Channels: a roadmap

Now consider the case where the users see channels with possibly different impulse responses $h_i(t)$ and therefore different associated eigenfunctions and eigenvalues. Since it is assumed that the communications interval is large relative the durations of all the channel impulse responses, the channel eigenfunctions will all be approximately sinusoidal "tonebursts".[2]

Under this assumption we then have

$$\mathbf{r} = \sum_{i=1}^{L} \Lambda_i^{1/2} \mathbf{S}_i \mathbf{b}_i + \mathbf{n} \tag{28}$$

which from the perspective of user k is

$$\mathbf{r} = \Lambda_k^{1/2} \mathbf{S}_k \mathbf{b}_k + \sum_{i=1, i \neq k}^{L} \Lambda_i^{1/2} \mathbf{S}_i \mathbf{b}_i + \mathbf{n} \tag{29}$$

For simplicity, assume all the Λ_i are invertible[3] The autocovariance of the received signal is then

$$E[\mathbf{r}\mathbf{r}^\top] = \sum_{i=1}^{L} \Lambda_i^{1/2} \mathbf{S}_i \mathbf{S}_i^\top \Lambda_i^{1/2} + N_0 \mathbf{I} \tag{30}$$

From the perspective of an individual user, the problem becomes how to select input waveforms for its bits in the presence of colored (although in this case correlated) "noise" consisting of both additive white Gaussian noise and interference from other users. Greedy interference avoidance performed by a single user will again produce WBE sequences for a fixed set of signatures employed by other users. Specifically, the signatures comprising $\mathbf{S}_k$ will be unit energy WBE sequences for the multiuser problem

---

[2] Special thanks are due to M. Honig for this perspective.

[3] Since in any event, no transmission energy will be placed in a signal dimension with sufficiently high attenuation, requiring the eigenvalue matrices to be invertible is for convenience only. For zero or near zero channel eigenvalues, the corresponding signal dimensions would simply be expurgated in the mathematical description.

$$\tilde{\mathbf{r}}_k = \Lambda_k^{-1/2}\mathbf{r} = \mathbf{S}_k\mathbf{b}_k + \sum_{j\neq k}\Lambda_k^{-1/2}\Lambda_j^{1/2}\mathbf{S}_j\mathbf{b}_j + \Lambda_k^{-1/2}N_0\mathbf{I} \tag{31}$$

This procedure could be iteratively applied by each user until some fixed point is (or is not) reached. We are currently investigating the relationship of such fixed points and sum capacity in this multiuser/multichannel problem. Our interest is particularly keen since formulations of this type figure prominently in other problems to which interference avoidance might be applied — most notably, multiple antenna systems.

6 Summary and Conclusion

Starting from a general signal space model of dispersive channels and using a sufficiently long transmission interval that intersymbol interference is essentially eliminated, we have shown that multiuser detection and channel equalization are (not surprisingly) isomorphic. Specifically, when M bits are to be transmitted by the user, we assign each bit a "CDMA" codeword where the ensemble meets the Welch bound with equality. Welch bound equality ensembles maximize user capacity and theoretically if used with Gaussian signaling, achieve channel capacity [6, 7].

The optimal codewords may be obtained through application of interference avoidance [8, 10, 11] or any other suitable numerical algorithm [6, 7]. Matched filter receivers for each bit are the optimal linear receiver and maximize the uniform SINR for each bit. We then extended (trivially) the method to the multiple user, identical channel case. Finally, we discussed the multiple user multiple channel case where different users preferentially occupy different portions of (possibly non-overlapping) signal spaces. Variants of this problem also arise when considering multiple antenna systems, or any configuration where users are constrained to different portions of the signal space, but where those spaces may partially overlap.

Notably absent from this work is a treatment of how the channel eigenfunctions and eigenvalues can be adaptively estimated, how covariance or codeword information is relayed from the receiver to the user(s), convergence speed, complexity issues and in general how such a theoretical concept can be made practical. To these concerns we respond *nolo contendre* at this point. Nevertheless, we feel that in light of increasingly versatile transmitters and receivers [17], interference avoidance offers an interesting window on multicarrier modulation and in general, transmission over dispersive channels.

References

[1] S. Kasturia, J.T. Aslanis, and J. M. Cioffi. Vector Coding For Partial Response Channels. *IEEE Tran. on Info. Th.*, 36(4):741–762, July 1990.

[2] J. W. Lechleider. The Optimum Combination of Block Codes and Receivers for Arbitrary Channels. *IEEE Tran. on Comm.*, 38(3):615–621, May 1990.

[3] R.G. Gallager. *Information Theory and Reliable Communication.* Wiley, 1968.

[4] J. L. Holsinger. Digital Communication Over Fixed-Time Dispersive Channels. *First IEEE Ann. Comm. Conv.*, pages 731–735, June 1965.

[5] G. W. Wornell. Spread-Signature CDMA: Efficient Multiuser Communication in the Presence of Fading. *IEEE Transactions on Information Theory*, 41(5), September 1995.

[6] P. Viswanath, V. Anantharam, and D. Tse. Optimal Sequences, Power Control and Capacity of Spread Spectrum Systems with Multiuser Linear Receivers. *IEEE Transactions on Information Theory*, June 1998. Submitted, revised. http://degas.eecs.berkeley.edu/~dtse/pub.html.

[7] P. Viswanath and V. Anantharam. Optimal sequences and sum capacity of synchronous cdma systems. College of Engineering UCB/ERL Memorandum 98/10, U.C. Berkeley, July 1998.

[8] S. Ulukus and R. D. Yates. Iterative signature adaptation for capacity maximization of cdma systems. In *Allerton Conf. on Comm., Control and Computing*, September 1998. See http://www.winlab.rutgers.edu/ ryates/papers/allerton98.ps.

[9] C. Rose, S. Ulukus, and R. Yates. Interference Avoidance in Wireless Systems. Winlab-TR 175, WINLAB, Rutgers University, 1999.

[10] C. Rose, S. Ulukus, and R. Yates. Interference Avoidance in Wireless Systems. *IEEE JSAC*, 1999. submitted 10/99, http://steph.rutgers.edu/ crose/papers/avoid11.ps.

[11] C. Rose. Convergence of Greedy Interference Avoidance. Winlab-TR 188, WINLAB, Rutgers University, 1999.

[12] H.L. Van Trees. *Detection, Estimation, and Modulation Theory, Part I*. Wiley, New York, 1968.

[13] Sergio Verdu. *Multiuser Detection*. Cambridge University Press, 1998.

[14] G.H. Hardy and J.E. Littlewood. *Inequalities: Theory of Majorization and its Applications*. Acadmic Press, 1979.

[15] T.M. Cover and J.A. Thomas. *Elements of Information Theory*. Wiley-Interscience, 1991.

[16] Gilbert Strang. *Linear Algebra and Its Applications*. Academic Press, second edition, 1992.

[17] Special issue on software radio. *IEEE Personal Communications Magazine*, 6(4), August 1999. Editors: K-C. Chen and R. Prasad and H.V. Poor.

Performance of optimal codes on Gaussian and Rayleigh fading channels : a geometrical approach.

Sandrine Vialle
Centre de Recherche Motorola
Espace Technologique
Commune de St Aubin
91193 Gif sur Yvette, France
vialle@com.enst.fr

Joseph Boutros
Communications Department
ENST University
46 Rue Barrault
75013 Paris, France
boutros@com.enst.fr

July 1, 1999

Abstract

The error probability of optimal codes with a given length n and a given rate R is estimated on additive white Gaussian noise channels with and without flat fading. A lower bound is given for the error probability on the Gaussian channel and a good approximation is derived in the presence of Rayleigh fading. The performance analysis is restricted to binary codes associated to a binary phase shift keying. However, the results can be easily generalized to any coded modulation.

1 Introduction

The channel coding theorem (e.g. see [8]) states that Gallager's exponent $E(R)$ is positive iff the code rate is smaller than capacity. Since the word error probability is upper bounded by $e^{-nE(R)}$, the latter can be used as an average performance estimation for the ensemble of random codes when n is finite and R has a fixed valid value. Instead of using the random coding technique, Shannon gave a lower bound [7] on the performance of all spherical codes in Gaussian channel based on a geometrical approach. Some modifications and improvements of his work have been done recently [1][5]. Nevertheless, the expression derived by Shannon for the main function $Q(\theta)$ is still the simplest one. This function gives the probability of moving outside a cone with half-angle θ. Its asymptotic expression is very accurate: for a code length greater than 100, it is too close to the exact optimal performance. This has been validated numerically by comparing $Q(\theta)$ to the upper bound derived by Shannon in the same paper [7].

In our paper, we generalize Shannon's geometrical approach to the Rayleigh fading channel. Section 3 summarizes the main results on the Gaussian channel and gives some important asymptotic expressions for the solid angle and the code rate. In section 4, we explain how to compute the conditional solid angle in the presence of independent flat fading. Finally, some numerical results are illustrated in the last section. The drawn curves can be used to check the performance of concatenated block and convolutional codes or any other type of codes capable of approaching channel capacity.

2 System model and notations

A linear binary code $[n,k]_2$ of length n and dimension k is considered. A codeword $c = (c_1, c_2, \ldots, c_n)$ is transmitted over an additive white gaussian noise (AWGN) channel using a binary phase shift modulation (BPSK). The symbols take the values $c_i = \pm\sqrt{2E_c}$, where E_c is the average energy per coded bit. The code rate is $R = k/n$ bits per dimension and the average energy per information bit is given by $E_b = E_c/R$.

Two channel models are studied: the Gaussian channel $y_i = c_i + z_i$ where z_i is a real Gaussian noise with zero mean and variance N_0, and the flat fading channel [6] $y_i = \alpha_i c_i + z_i$ where $\alpha_i \in [0, +\infty[$ is Rayleigh distributed with probability density function $p(\alpha_i) = 2\alpha_i e^{-\alpha_i^2}$. Note that $E[\alpha_i^2] = 1$ so the average energy value is not changed by the Rayleigh channel.

Let us now make a geometrical representation of the code set in the n-dimensional real space $\mathbb{R}^n$. We assume that the codewords are equally probable and selected at random from the ensemble $\{\pm\sqrt{2E_c}\}^n$. Thus, the codewords c are uniformly distributed on the n-dimensional sphere $Sph(O,n)$ centered at the origin with squared radius $E = 2nE_c$. The total space solid angle is denoted Ω_n. The solid angle of a circular cone with half angle θ is denoted $\Omega(\theta)$. The codeword c is represented as a point lying on $Sph(O,n)$ and corresponding to a solid angle $\Omega(\theta_0) = \Omega_n/M$, where $M = 2^{nR}$ is the code size and $R = R(\theta_0)$.

Figure 1 shows the codeword c surrounded by a circular cone $C(c)$ delimited by an $n-1$-dimensional sphere (circle) $Sph(a, n-1)$ and a pyramid $V(c)$ drawn in an hexagonal form. $V(c)$ is the Voronoi region associated to the codeword c, i.e. $V(c)$ is the decision region of c when maximum likelihood (ML) decoding is applied. The circle $Sph(a, n-1)$ is the intersection of the sphere $Sph(O,n)$ and a hyperplane denoted by $\mathcal{H}$. The radius $\rho_0 = \rho_0(\theta_0)$ of the circle is chosen such that the solid angle $\Omega(\theta_0)$ of the pyramid $V(c)$ is equal to that of the cone $C(c)$ delimited by $Sph(a, n-1)$. Notice that the projection of the codeword point c on the hyperplane $\mathcal{H}$ is denoted a. Two other radii are also shown on figure 1, the packing radius ρ_p and the covering radius ρ_c of the spherical code equivalent to the considered linear binary code. The vector modules of c and a are related by $||a|| = ||c|| \cos(\theta_0)$ where $||c||^2 = E$.

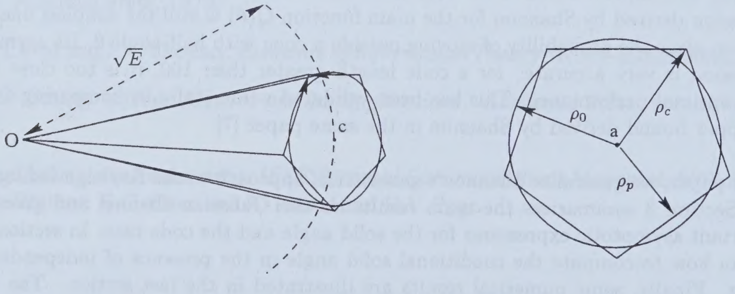


Figure 1: Geometrical representation of the solid angle associated to a codeword.

3 Optimal performance on the AWGN channel

The codewords are supposed to be uniformly distributed and have the same a priori probability. By symmetry, we can state that the average word error probability P_{ew} is equal to the conditional error probability $P_e(c)$ when c is transmitted, e.g. $c = (+\sqrt{2E_c}, +\sqrt{2E_c}, \ldots, +\sqrt{2E_c})$. The error probability $P_e(c)$ is thus the probability for the received signal to be moved by the Gaussian noise z outside the pyramid $V(c)$.
For all n, the Gaussian noise probability density is a decreasing function of the Euclidean distance d:

$$\frac{\partial p_z}{\partial V} = \frac{1}{(2\pi N_0)^{n/2}} \exp(-\frac{d^2}{2N_0}) \tag{1}$$

It can be easily shown that

$$\int_{C(c)} \frac{\partial p_z}{\partial V} dV \geq \int_{V(c)} \frac{\partial p_z}{\partial V} dV \tag{2}$$

Let us denote $Q(\theta)$ the probability for a codeword on the axis of a cone of solid angle $\Omega(\theta)$ being moved outside this cone. We have

$$Q(\theta_0) = 1 - \int_{C(c)} \frac{\partial p_z}{\partial V} dV$$

Using inequality 2 and the above definition of $Q(\theta_0)$, we obtain the following lower bound for the word error probability [7]

$$Q(\theta_0) \leq P_{ew}$$

In the following, we explain how to compute $Q(\theta_0)$ for a given length n and a given rate R. The relation $R = R(\theta_0)$ and the expression of $Q(\theta)$ have been developed by C.E. Shannon [7] and may also be applied to any spherical code [3], i.e. any coded modulation with equal energy signals.
Let us denote $\Gamma(x)$ the Euler function defined by $\Gamma(x) = \int_0^{+\infty} e^{-t} t^{x-1} dt$. The volume of an n-dimensional sphere with unit radius is [3]

$$Vol(n) = \frac{\pi^{n/2}}{\Gamma\left(\frac{n}{2}+1\right)}$$

For large space dimensions, $n >> 1$, the Stirling approximation yields

$$Vol(n) \approx \frac{\left(\frac{2e\pi}{n}\right)^{n/2}}{\sqrt{n\pi}}$$

The total solid angle Ω_n is equal to the surface of the n-dimensional sphere considered above and is related to its volume by $\Omega_n = nVol(n)$. The solid angle $\Omega(\theta)$ is computed by summing elementary ring surfaces $d\Omega(\phi) = \Omega_{n-1} \sin^{n-2} \phi d\phi$.

$$\Omega(\theta) = \frac{(n-1)\pi^{\frac{n-1}{2}}}{\Gamma\left(\frac{n+1}{2}\right)} \int_0^{\theta} \sin^{n-2} \phi d\phi$$

Now the code rate is related to the half angle by $M = 2^{nR} = \frac{\Omega_n}{\Omega_0}$.

$$R = \frac{1}{n} \log_2 \left(\frac{\sqrt{\pi}\Gamma\left(\frac{n-1}{2}\right)}{\Gamma\left(\frac{n}{2}\right) \int_0^{\theta_0} \sin^{n-2} \phi d\phi} \right)$$

Asymptotically, when $n >> 1$, the expressions of the solid angle and the rate become

$$\Omega(\theta) \approx \frac{1}{\pi\sqrt{2}}\left(\frac{2e\pi}{n}\right)^{n/2}\frac{\sin^{n-1}\theta}{\cos\theta} \tag{3}$$

$$2^{nR} \approx \frac{\sqrt{2\pi n}\sin\theta_0\cos\theta_0}{\sin^n\theta_0} \tag{4}$$

Finally let us compute $Q(\theta)$ from its derivative

$$Q(\theta) = \int_\theta^{\frac{\pi}{2}} -dQ(\phi) + Q\left(\frac{\pi}{2}\right) \tag{5}$$

Notice that $Q\left(\frac{\pi}{2}\right) = \frac{1}{2}erfc\left(\frac{E}{2N_0}\right)$, where $erfc$ is the complementary error function [6]. This term will be neglected since it has a minor influence on $Q(\theta)$.
Consider equation 1 and figure 2. dV is the volume of a ring given by its depth dr, its width $rd\theta$ and placed at a distance d from c given by $d^2 = E + r^2 - 2r\sqrt{E}\cos\theta$. The cone element of volume dV is

$$dV = rdrd\theta\frac{(n-1)\pi^{\frac{n-1}{2}}(r\sin\theta)^{n-2}}{\Gamma(\frac{n+1}{2})}$$

The probability density for moving c inside this cone element according to the noise Gaussian distribution is

$$-d^2Q(r,\theta) = \frac{1}{(2\pi N_0)^{n/2}}\exp\left(-\frac{E+r^2-2r\sqrt{E}\cos\theta}{2N_0}\right)\frac{(n-1)\pi^{\frac{n-1}{2}}(r\sin\theta)^{n-2}}{\Gamma\left(\frac{n+1}{2}\right)}rdrd\theta$$

The differential $-dQ(\theta)$ is obtained by integrating $-d^2Q(r,\theta)$ on the variable r, between 0 and $+\infty$:

$$-\frac{dQ(\theta)}{d\theta} = \frac{(n-1)\exp\left(-\frac{E\sin^2\theta}{2N_0}\right)\sin^{n-2}\theta d\theta}{(2N_0)^{n/2}\sqrt{\pi}\Gamma\left(\frac{n+1}{2}\right)}\int_0^{+\infty}\exp\left(-\frac{(r-\sqrt{E}\cos\theta)^2}{2N_0}\right)r^{n-1}dr$$

This equation can be approximated asymptotically [7]

$$-dQ(\theta) \approx \frac{(n-1)}{\sqrt{n\pi}\sqrt{1+G^2\sin^2\theta}}\left[G\sin\theta\exp\left(-\frac{E_c}{N_0}+\frac{1}{2}\sqrt{\frac{2E_c}{N_0}}G\cos\theta\right)\right]^n d\theta \tag{6}$$

where $G = \frac{1}{2}\left[\sqrt{\frac{2E_c}{N_0}}\cos\theta + \sqrt{\frac{2E_c}{N_0}\cos^2\theta + 4}\right]$. To derive the numerical results versus the signal-to-noise ratio E_b/N_0, we used equations (5) and (6) combined together with the relation $E_c = RE_b$ to evaluate the lower bound on the word error probability

$$Q(\theta_0) \approx \frac{1}{\sqrt{n\pi}}\frac{1}{\sqrt{1+G^2}\sin\theta_0}\frac{\left[G\sin\theta_0\exp\left(-\frac{E_c}{N_0}+\frac{1}{2}\sqrt{\frac{2E_c}{N_0}}G\cos\theta_0\right)\right]^n}{\sqrt{\frac{2E_c}{N_0}}G\sin^2\theta_0-\cos\theta_0} \tag{7}$$

The cone half-angle θ_0 is computed by solving equation (4) for a fixed space dimension n and a fixed rate R.

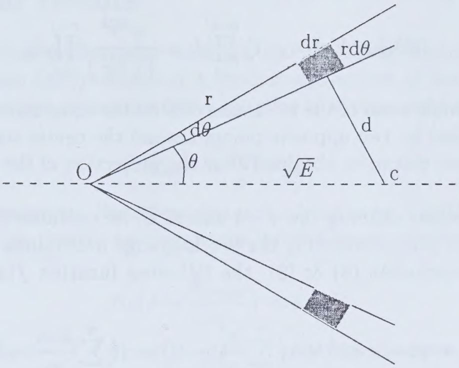


Figure 2: Section of the cone of half-angle θ.

4 Optimal performance on the Rayleigh channel

The spherical code is affected by a real fading $\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_n)$. The codeword point c is transformed into $\alpha c = (\alpha_1 c_1, \alpha_2 c_2, \ldots, \alpha_n c_n)$ and then corrupted by the white Gaussian noise. The symmetric Voronoi regions $V(c)$ of the spherical code without fading are now transformed into $\alpha V(c)$ which is not completely superimposed with the exact Voronoi region $V(\alpha c)$ of the faded spherical code. The problem of finding the shape and boundaries of $V(\alpha c)$ is intractable especially in high dimensions. Hence, we are only able to find theoretically an approximation (not an exact lower bound) for the word error probability of the linear binary code $[n, k]_2$ over the Rayleigh channel. To do so, we first approximate the solid angle of $V(\alpha c)$ by that associated to $\alpha C(c)$. Then we compute the equivalent angle $\theta_1 = \theta_1(\alpha. \theta_0)$ corresponding to a circular cone having the same solid angle and finally we evaluate the conditional error probability $P_{ew}(\theta_1) \approx Q(\theta_1)$.

According to section 2, the elliptic cone $\alpha C(c)$ is now delimited by the $n-1$-dimensional ellipsoid $\mathcal{E} = \alpha Sph(a, n-1)$. The latter is defined as the intersection of the new hyperplane $\alpha\mathcal{H}$ and the n-dimensional ellipsoid centered at the origin $\alpha Sph(O, n)$. All points x belonging to $\mathcal{E}$ satisfy the two following equations

$$\alpha Sph(O, n): \qquad \sum_{i=1}^{n} \frac{x_i^2}{\alpha_i^2} = E \tag{8}$$

and

$$\alpha\mathcal{H}: \qquad \sum_{i=1}^{n} \frac{a_i x_i}{\alpha_i} = ||a||^2 \tag{9}$$

It is easy to show that the point αa is the center of the ellipsoid $\mathcal{E}$ and that αa belongs to the line $O - \alpha c$.
The solid angle $\Omega(\theta_1)$ of $\alpha C(c)$ is equal to the surface of $\mathcal{E}$ (i.e. $n-1$-dimensional volume)

projected on the n-dimensional unity sphere. The surface $\Omega(\theta_1)$ is well approximated by

$$\Omega(\theta_1) \approx Vol(n-1) \prod_{j=1}^{n-1} d_j = \frac{\pi^{\frac{n-1}{2}}}{\Gamma\left(\frac{n+1}{2}\right)} \prod_{j=1}^{n-1} d_j \tag{10}$$

where $\{d_j\}$ are the projections of the $n-1$ axes of $\mathcal{E}$ on the unity sphere. Notice that each ellipsoid axis is defined by two opposite points around the center αa. The distance d_j is the smallest of the two distances obtained after the projection of the two opposite points.

To find the $2n-2$ points defining the $n-1$ axis of $\mathcal{E}$, we optimize $||x - \alpha\ a||^2$ under the two constraints (8) & (9). Introducing the two Lagrange multipliers λ and γ respectively associated to the constraints (8) & (9), the following function $f(x)$ is optimized with respect to x

$$f(x) = ||x - \alpha\ a||^2 - \lambda\Big(\sum_{i=1}^{n} \frac{x_i^2}{\alpha_i^2} - E\Big) - \gamma\Big(\sum_{i=1}^{n} \frac{a_i x_i}{\alpha_i} - ||a||^2\Big) \tag{11}$$

The solutions to the above optimization (algebraic manipulations and details omitted) are the $2n-2$ points $x = (x_1, x_2, \ldots, x_n)$ given by

$$x_i = \frac{\frac{\gamma}{2} + \alpha_i^2}{\alpha_i^2 - \lambda} \times \alpha_i a_i \qquad i = 1, \ldots, n \tag{12}$$

The multiplier λ takes $n-1$ values which are the solutions of the equation

$$\sum_{i=1}^{n} \frac{1}{\alpha_i^2 - \lambda} = 0$$

For each value of λ, there are two values for γ associated to the two opposite points around αa and given by the following second degree equation

$$V_1\gamma^2 + 4V_2\gamma + 4\left(V_4 - \frac{1}{\cos^2\theta_0}\right) = 0$$

where

$$\begin{aligned} V_1 &= \sum_{i=1}^{n} \frac{1}{(\alpha_i^2 - \lambda)^2} \\ V_2 &= \sum_{i=1}^{n} \frac{\alpha_i^2}{(\alpha_i^2 - \lambda)^2} \\ V_4 &= \sum_{i=1}^{n} \frac{\alpha_i^4}{(\alpha_i^2 - \lambda)^2} \end{aligned}$$

The distances $\{d_j\}$ needed in equation (10) are defined as

$$d_j = ||\frac{x}{||x||} - \frac{\alpha a}{||\alpha a||}|| \qquad j = 1...n-1$$

where the point x is given by (12) and the j^{th} value of λ and γ,

$$\gamma = \frac{-2V_2 + 2\sqrt{V_2^2 - V_1(V_4 - 1/\cos^2\theta_0)}}{V_1}$$

To summarize, for a given fading $\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_n)$, evaluate the equivalent solid angle $\Omega(\theta_1)$ using (10), find θ_1 by solving (3) and then compute the conditional error probability $Q(\theta_1)$ from equation (7).

5 Numerical results

Notice that the optimal performance analysis, i.e. the lower bound $Q(\theta_0)$ of section 3 and the approximation $Q(\theta_1)$ of section 4, give an estimation of the optimal word error probability versus the signal-to-noise ratio. In order to estimate the bit error probability, we consider

$$P_{eb} \approx \frac{d_{Hmin}}{n} P_{ew}$$

where d_{Hmin} is the minimum Hamming distance of the code. Gallager [4] showed that when n is large enough, binary random codes quickly approach the Gilbert-Varshamov bound, i.e.

$$H_2\Big(\delta = \frac{d_{Hmin}}{n}\Big) = 1 - R$$

where $H_2(x) = -x \log x - (1-x)\log(1-x)$ is the entropy function, and $0 < \delta \leq \frac{1}{2}$. Consequently

$$P_{eb} \approx H^{-1}_{2|[0,1/2]}(1-R) \times P_{ew} \tag{13}$$

Typical values of δ are listed in the table below

R	$\delta = H^{-1}_{2\|[0,1/2]}(1-R)$
1/4	0.2145
1/2	0.1100
3/4	0.0417

The bit error rate performance of an optimal code is compared to that of a parallel concatenated convolutional code (Turbo code [2]) in figure 3. The code rate is 0.5 bits per dimension. The turbo code constituent is the recursive systematic convolutional code defined by the octal generators 23, 35. The number of decoding iterations is 12 for the length $n = 2048$ (interleaver size 1024) and 20 decoding iterations for the length $n = 131072$ (interleaver size 65536). As illustrated, the turbo code is 0.9 dB away from the optimal code at 10^{-5} when $n = 2048$. The gap is about 0.6 dB at 10^{-4} for the longer code $n = 131072$.

Finally, figures 4, 5 & 6 show the optimal bit error probability for a binary code of rate 1/4, 1/2 and 3/4 respectively. The curves are plotted for four different lengths $n = 100$, 200, 800, 2000 bits on both Gaussian and Rayleigh fading channels. The curves on the fading channel exhibit a very high diversity since the minimum Hamming distance of the random code derived from the Gilbert-Varshamov bound is relatively large. For the greatest code length $n = 2000$, the gap between the Rayleigh and the Gaussian channels is about 2.25, 2.50 & 2.75 dB with the code rates 0.25, 0.50 & 0.75 bits/dim respectively.

References

[1] G. Battail, H. Magalhaes de Oliveira : "Probabilité d'erreur du codage aléatoire avec décodage optimal sur le canal à bruit gaussien additif affecté ou non de fluctuations d'amplitude,", *Annales des Télécommunications* vol.48, no 1-2, pp. 15-28, 1993.

[2] C. Berrou, A. Glavieux, P. Thitimajshima: "Near Shannon Limit Error-Correcting Coding and Decoding: Turbo-Codes," *Proc. of ICC'93, Geneva* pp. 1064-1070, 23-26 May 1993.

[3] J.H. Conway, N.J. Sloane, *Sphere packings, lattices and groups*, 3rd edition, 1998, Springer-Verlag, New York.

[4] R.G. Gallager: Low-density parity-check codes, MIT Press, 1963.

[5] D.E. Lazic, T. Beth, M. Calic: "How close are turbo codes to optimal codes?," *Proc. Int. Symp. on Turbo Codes*, pp. 192–195, Brest, September 1997.

[6] John G. Proakis, *Digital Communications*, 3rd edition, McGraw-Hill, 1995.

[7] C.E. Shannon: "Probability of Error for Optimal Code in Gaussian Channel," *The Bell System Technical Journal*, vol. 38, no.3, May 1959.

[8] A.J. Viterbi, J.K. Omura: Principles of Digital Communication and Coding, McGraw-Hill Book Company, 1979.

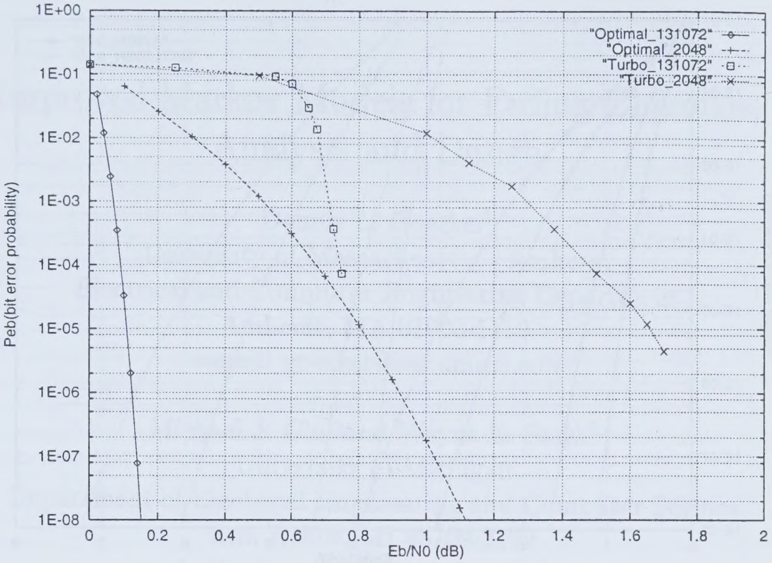


Figure 3: Turbo codes versus optimal codes, $n = 131072$ & 2048, rate 1/2

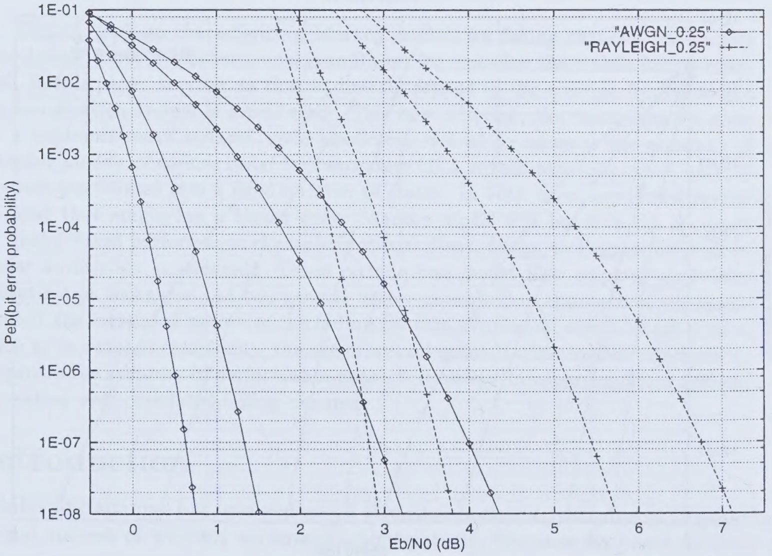


Figure 4: Optimal performance, rate 1/4, code length $n = 100, 200, 800, 2000$

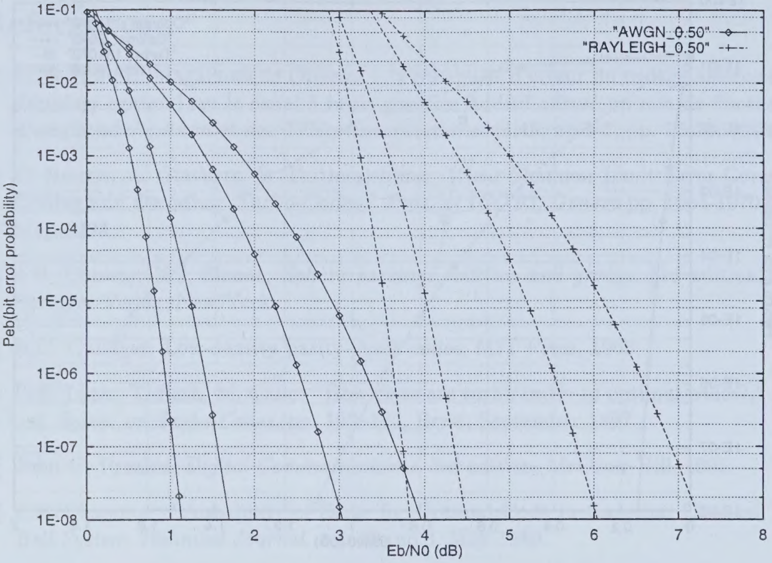


Figure 5: Optimal performance, rate 1/2, code length $n = 100, 200, 800, 2000$

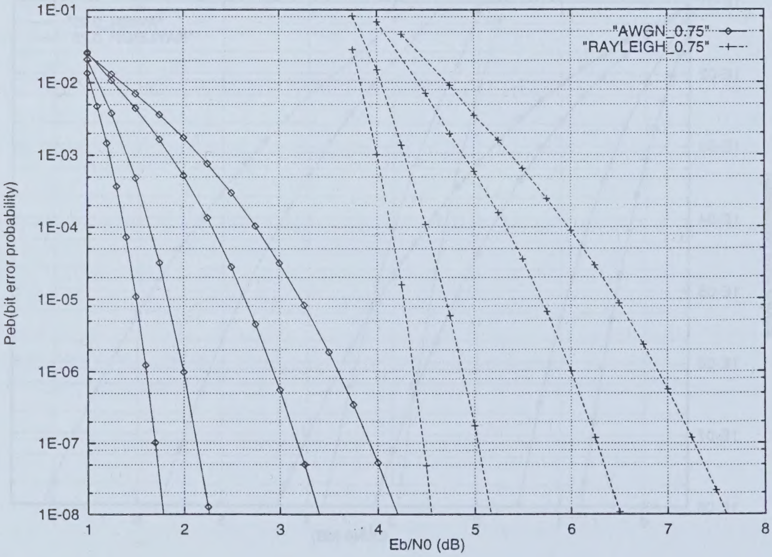


Figure 6: Optimal performance, rate 3/4, code length $n = 100, 200, 800, 2000$

Improved Markov Models for Fading Channels: Analysis and Design

Dennis L. Goeckel[1]
University of Massachusetts - Amherst
Electrical and Computer Engineering Department
Amherst, MA 01003-5110
e-mail: goeckel@ecs.umass.edu

Michael J. Chu and Wayne E. Stark[2]
University of Michigan
Department of Electrical Engineering and Computer Science
Ann Arbor, MI 48109-2122
{mikechu,stark}@eecs.umich.edu

Abstract

Recent work on characterizing Markov models for fading channels has largely concluded that a first-order model is sufficient for modeling the mobile fading channel. In this paper, a criterion that is directly related to the error performance of a communication system is introduced. This new criterion, the mean-squared error of a minimum-mean squared error predictor, is used to measure the accuracy of Markov models of various order for two cases: (1) a continuous state space, and (2) a space partitioned into a fixed number of states. In both cases, numerical results suggest that employing a higher-order Markov model will improve the accuracy of model-based performance characterizations. Next, design techniques for higher-order models are considered. These include one model that employs a product quantization technique and two general vector quantization approaches, which can exploit the correlated nature of the fading process. Numerical results suggest that, with no increase in complexity, the use of second-order Markov models significantly improves the accuracy of performance characterizations for communication systems operating over correlated fading channels.

1 Introduction

The mobile radio channel has received much attention in recent years due to the widespread commercial success of wireless communication systems. Because an exact analysis of a

---

[1] This paper is based in part upon work supported by the National Science Foundation under Grant NCR-9714597.

[2] This research is supported in part by the Department of Defense Research and Engineering Multidisciplinary University Research Initiative on "Low Energy Electronics Design for Mobile Platforms" and managed by the Army Research Office under Grants DAAH04-96-1-0377 and DAAH04-96-1-0110.

communication system operating over the mobile radio channel is often prohibitively complex, there is a critical need for accurate models of channels with memory. These models are needed not only to analyze traditional coded systems, but more importantly, the use of accurate channel models will become an integral component in decoders which seek to achieve high-rate communications by avoiding the capacity loss due to interleaving [1].

Much of previous research has focused on the design of first-order Markov models for the mobile radio channel. Perhaps the most commonly used model is a birth-death process, where the state transitions are generated according to level crossing rates [2]. The authors in [3] validate the use of first-order Markov models by using an information theoretic argument, and numerous references can be found in the literature that employ these first-order models. However, recent authors (e.g. [4]) have questioned whether the first-order Markov model accurately represents the mobile radio channel.

In this paper, analytical and numerical results are presented which suggest that the accuracy of model-based performance characterizations can be significantly improved with no increase in complexity by employing Markov models of order greater than one. A significant step in the approach is the derivation of a new criterion - the mean-squared error of a minimum mean-squared error (MMSE) one-step predictor based on the model under consideration; it is shown that this criterion is directly related to the error probability of a representative system and thus serves as an alternative to the criterion of [3]. This new criterion will suggest that higher-order models can provide significant gains in accuracy over first-order Markov models when each employs a continuous state space. Asymptotic quantization theory is then used to predict more moderate gains when both models employ an equal, finite number of states to partition their state spaces. Having motivated the need for higher-order models, design techniques for higher-order Markov models and associated numerical results are presented which support the analysis.

The structure of the paper is as follows. Section 2 describes the channel model. In Section 3, the new criterion for model quality is derived, and this criterion is used in Section 4 to evaluate Markov models of various orders. Section 5 presents three design techniques for second-order Markov models and the associated numerical results. Finally, section 6 summarizes continuing work.

2 Channel Model

To isolate the effect of correlation in time of the fading process, a frequency nonselective, time-selective fading model will be assumed. The fading is then represented as a complex multiplier $X(t)$ applied to the transmitted signal. The Gaussian wide-sense stationary uncorrelated scattering (GWSSUS) model will be assumed; thus, $X(t) = X_R(t) + jX_I(t)$ is complex Gaussian, where $X_R(t)$ and $X_I(t)$ are the independent real and imaginary components of $X(t)$, respectively, $j = \sqrt{-1}$, $E[X_R(t)] = E[X_I(t)] = 0$, and $E[X_R(t)X_R(t+\tau)] = E[X_I(t)X_I(t+\tau)] = K_X(\tau)$. Throughout this work it will generally be assumed that $K_X(\tau) = J_0(2\pi f_d \tau)$, where $J_0(\cdot)$ is the zeroth-order Bessel function, and f_d is the Doppler frequency. With pulse shaping that results in no intersymbol interference, a discrete-time system model is given by:

$$r_n = X_n s_n + N_n$$

where $\mathbf{r} = (r_0, r_1, r_2, \ldots)$ is a sequence of T-spaced samples of the receiver matched filter output, $\mathbf{s} = (s_0, s_1, s_2, \ldots)$ is the sequence of transmitted symbols, $\mathbf{N} = (N_0, N_1, N_2, \ldots)$ is an independent and identically distributed (IID) sequence of zero-mean complex Gaussian random variables with independent in-phase and quadrature components, each with variance $\frac{N_0}{2}$, and T is the symbol period. It will be assumed that the channel fades slowly enough such that it is constant over a symbol interval and thus $X_n = X_{R,n} + jX_{I,n} = X(nT_s)$. Finally, without loss of generality, the channel will be normalized such that $E[X_R(t)^2] = E[X_I(t)^2] = 1$.

3 Prediction Error as a Measure of Model Quality

With the definition $R_i \triangleq |X_i|$, [3] defines the additional accuracy gained by employing a second-order Markov model for the discrete-time process $\mathbf{R} = (R_0, R_1, R_2, \ldots)$ as the ratio:

$$\frac{I(R_i; R_{i-2}|R_{i-1})}{I(R_i; (R_{i-1}, R_{i-2}))}$$

where $I(R_i; R_{i-2}|R_{i-1})$ is the mutual information between R_i and R_{i-2} given R_{i-1}, and $I(R_i; (R_{i-1}, R_{i-2}))$ is the mutual information between R_i and the pair (R_{i-1}, R_{i-2}). Using this criterion, it is concluded that the first-order Markov model is sufficient to approximate the fading channel. Although this criterion appears reasonable, it is not clear how it is tied to the error probability of a system operating over the mobile channel. Throughout this paper, the mean-squared error of a minimum mean-squared error predictor of the next sample of a process derived from $\mathbf{X}$ given the previous m samples of the process is used as the criterion by which the quality of an m^{th}-order model is judged. The justification for this criterion is given in this section.

For a system that employs error control coding over a memoryless channel, the pairwise error probability between any two codewords separated by Hamming distance L can be obtained as the probability of error of an $(L, 1)$ repetition code on the same channel. When the channel is not memoryless, the same relation does not hold; however, considering the probability of codeword error of a repetition code will yield insight into how the order of a Markov model directly affects performance characterizations. Assume that the coded bits of a rate $\frac{1}{L}$ repetition code are each mapped to an antipodal signal set $\{-\sqrt{E_s}, \sqrt{E_s}\}$, where E_s is the transmitted energy per coded bit, to yield the transmitted vector $\mathbf{s}_L = (s_0, s_1, \ldots, s_{L-1})^T = s_0(1, 1, \ldots, 1)^T$, which is affected by correlated fading. Let $\mathbf{X}_L = (X_0, X_1, \ldots, X_{L-1})^T = \mathbf{X}_{R,L} + j\mathbf{X}_{I,L}$ and $K_{\mathbf{X}_L} \triangleq E[\mathbf{X}_{R,L}\mathbf{X}_{R,L}^T] = E[\mathbf{X}_{I,L}\mathbf{X}_{I,L}^T]$. Thus, $\mathbf{X}_L$ is a complex Gaussian random vector with covariance matrix of *either* the in-phase or quadrature portions given by $K_{\mathbf{X}_L}$. The received vector is given by $\mathbf{r}_L = s_0\mathbf{X}_L + \mathbf{N}_L$. Assuming maximum ratio combining with perfect phase and channel estimation, the analysis follows that of [5] to yield the probability of codeword error as [6]:

$$\begin{aligned} P_e &= P\left(\mathrm{Re}\{\mathbf{X}_L^H\mathbf{X}_L\sqrt{E_s} + \mathbf{X}_L^H\mathbf{N}_L\} < 0 \mid s_0 = \sqrt{E_s}\right) \\ &= E_{\mathbf{X}_L}\left[Q\left(\sqrt{\frac{2\mathbf{X}_L^H\mathbf{X}_L E_s}{N_0}}\right)\right] \\ &\leq \frac{1}{2}E_{\mathbf{X}_L}\left[e^{-\mathbf{X}_L^H\mathbf{X}_L\frac{E_s}{N_0}}\right] \\ &= \frac{1}{2\left|I + \frac{2E_s}{N_0}K_{\mathbf{X}_L}\right|} \end{aligned}$$

Let $\mathbf{Z} \triangleq \sqrt{\frac{2E_s}{N_0}}\mathbf{X}_R + \mathbf{U}$, where $\mathbf{U}$ is an IID sequence of zero-mean unit-variance Gaussian random variables that is independent of $\mathbf{X}$. Then, using methods suggested in Chapter 16 of [7],

$$\begin{aligned} P_e &\leq \frac{1}{2}(2\pi e)^L \; 2^{-2h(\mathbf{Z}_L)} \\ &= \frac{1}{2}(2\pi e)^L \; 2^{-2\sum_{l=0}^{L-1} h(Z_l|Z_0,Z_1,\ldots,Z_{l-1})} \\ &= \frac{1}{2}(2\pi e)^L \; 2^{-2\sum_{l=0}^{L-1} \frac{1}{2}\log_2\left(2\pi e\sigma^2_{\mathbf{Z},l}\right)} \\ &= \frac{1}{2}\prod_{l=0}^{L-1}\frac{1}{\sigma^2_{\mathbf{Z},l}} \end{aligned}$$

where $h(\mathbf{Z}_L)$ is the joint differential entropy [7, pg. 230] of $\mathbf{Z}_L$, and $\sigma^2_{\mathbf{Z},l}$ is the mean-squared error of an MMSE predictor of Z_l given $(Z_0, Z_1, \ldots, Z_{l-1})$.

To consider how the above analysis impacts the choice of model order, let $\tilde{P}_e$ be the right hand side of the equation above and $\tilde{P}_{e,m}$ be an analogous upper bound to the probability of error of the repetition code obtained when the process $\mathbf{X}$ is assumed to be m^{th}-order Markovian. Under the assumption that $\mathbf{X}$ is m^{th}-order Markovian, $\mathbf{Z}$ is approximately m^{th}-order Markovian for reasonably large α, and

$$\frac{\tilde{P}_e}{\tilde{P}_{e,m}} \approx \prod_{l=m+1}^{L-1}\frac{\sigma^2_{\mathbf{Z},m}}{\sigma^2_{\mathbf{Z},l}}$$

and the improvement in the bound on error probability if $\mathbf{X}$ is assumed to be Markov of order $m+1$ rather than Markov of order m is approximated as

$$\frac{\tilde{P}_{e,m+1}}{\tilde{P}_{e,m}} \approx \left(\frac{\sigma^2_{\mathbf{Z},m}}{\sigma^2_{\mathbf{Z},m+1}}\right)^{L-m-1}. \tag{1}$$

Thus, since $\sigma^2_{\mathbf{Z},m} \geq \sigma^2_{\mathbf{Z},m+1}$, there is a direct relationship between reducing the mean-squared error of an MMSE predictor of $\mathbf{Z}$ and improving the upper bound to the error probability of a repetition code.

4 Accuracy of Markov Models

Motivated by (1), the mean-squared error of a one-step MMSE predictor based on the sequence $\mathbf{Z} = \alpha\mathbf{X}_R + \mathbf{U}$ is used to evaluate the accuracy of Markov models of different orders. The constant α depends on the signal-to-noise ratio (SNR), and $\mathbf{X}_R$ and $\mathbf{U}$ are defined as in the previous section.

4.1 No Quantization

Consider the quantity $\sigma^2_{\mathbf{Z},m}$, the mean-squared error of an MMSE predictor of Z_m based on $\mathbf{Z}_m \triangleq (Z_0, Z_1, Z_2, \ldots, Z_{m-1})^T$. Because Z_m and the random variables in $\mathbf{Z}_m$ are jointly Gaussian, the MMSE predictor is linear and is given by:

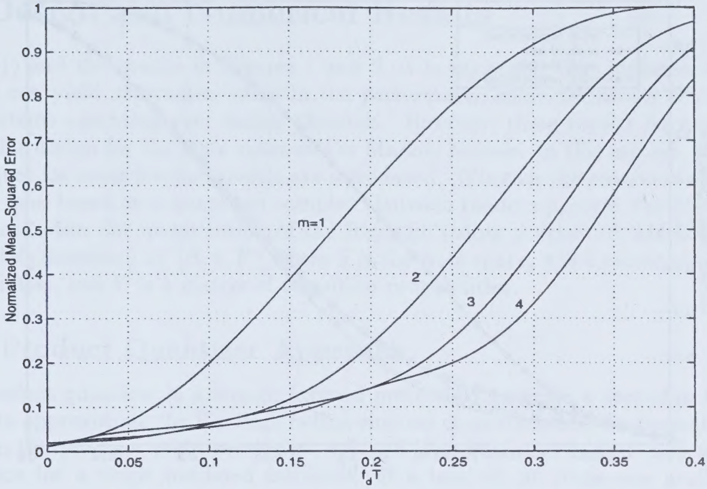


Figure 1: Normalized mean-squared error performance of predictors of various order m as a function of f_dT for $\alpha = 10$.

$$\begin{aligned}\hat{Z}_m &= E[Z_m\mathbf{Z}_m^T](E[\mathbf{Z}_m\mathbf{Z}_m^T])^{-1}\mathbf{Z}_m \\ &= \alpha^2\mathbf{k}_{\mathbf{X}_m}^T\left(\alpha^2 K_{\mathbf{X}_m} + I\right)^{-1}\mathbf{Z}_m\end{aligned}$$

where $\mathbf{k}_{\mathbf{X}_m} = E[X_{R,m}\mathbf{X}_{R,m}]$, and, recalling that the channel is normalized to $E[X_{R,n}^2] = E[X_{I,n}^2] = 1$,

$$\begin{aligned}\sigma_{\mathbf{Z},m}^2 &= E[Z_m^2] - E[Z_m\mathbf{Z}_m^T](E[\mathbf{Z}_m\mathbf{Z}_m^T])^{-1}E[Z_m\mathbf{Z}_m] \\ &= \alpha^2 + 1 - \alpha^4\mathbf{k}_{\mathbf{X}_m}^T\left(\alpha^2 K_{\mathbf{X}_m} + I\right)^{-1}\mathbf{k}_{\mathbf{X}_m}\end{aligned}$$

In Figure 1, the normalized mean-squared error of the MMSE predictor is plotted as a function of f_dT for various m when $\alpha = 10$. The mean-squared error is normalized by its maximum value, $\alpha^2 + 1$. At low f_dT where the channel is changing slowly, all models perform equally well. The error floor is due to the prediction of a noisy process based on noisy measurements. As the channel changes more quickly, the higher-order models exhibit significantly better mean-squared error performance, but the incremental gain decreases as order increases. In contrast to the criterion of [3], these results suggest a significant improvement in the characterization of system performance by going to higher-order Markov models.

4.2 Quantized Process

The previous results indicate that the fading process is not first-order Markovian with respect to the mean-squared error of the MMSE predictor of the next sample. However, when a finite-state Markov chain model is used, increasing the order implies spreading the

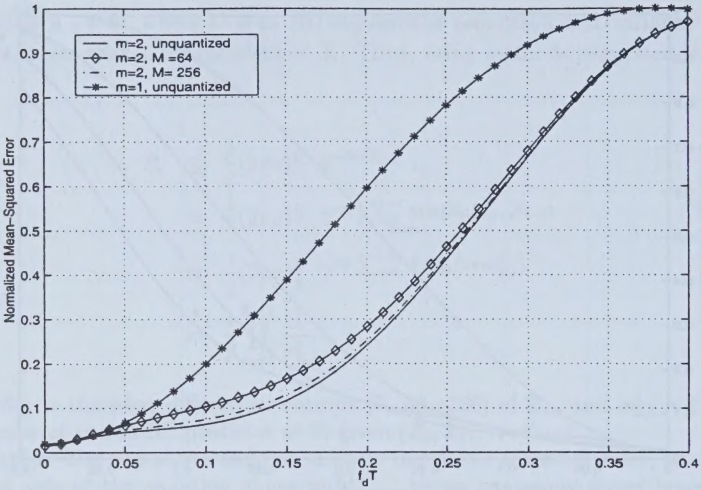


Figure 2: Normalized mean-squared error performance of second-order predictors for various M as a function of $f_d T$ for $\alpha = 10$.

states of the chain over a space of much larger dimensionality. Thus, it is imperative that the effect of quantization noise be included in the analysis of the quality of an m^{th}-order model.

Consider the quantity $\sigma^2_{\mathbf{Z},m,M}$, the mean-squared error of a MMSE linear predictor of Z_m from $\mathbf{Z}_{m,M} = \alpha(\mathbf{X}_m + \mathbf{Q}_m) + \mathbf{U}_m$, which employs a quantized version of $\mathbf{X}_m$. Here an m-dimensional quantizer with M quantization points is employed, and $\mathbf{Q}_m$ is the m-dimensional quantization error vector. If the quantizer rate is high, the components of $\mathbf{Q}_m$ can be assumed to be IID– this is assumed throughout this work. Note that $\mathbf{Q}_m$ is not a Gaussian vector in general, and thus the error considered in this section is the mean-squared error of the MMSE *linear* predictor and not the error of the (unconstrained) MMSE predictor. Following the previous results,

$$\sigma^2_{\mathbf{Z},m,M} = \alpha^2 + 1 - \alpha^4 \mathbf{k}^T_{\mathbf{X}_m} \left(\alpha^2(K_{\mathbf{X}_m} + \sigma^2_{Q,m,M} I) + I\right)^{-1} \mathbf{k}_{\mathbf{X}_m}$$

where $\sigma^2_{Q,m,M}$ is the mean-squared error per dimension of a vector quantizer of $\mathbf{X}_m$ dimensions with M quantization points. For an optimal quantizer, asymptotic quantization theory yields [8]:

$$\sigma^2_{Q,m,M} = \begin{cases} \frac{\sqrt{3}\pi}{2}\frac{1}{M^2}, & m = 1 \\ \frac{2\pi h^*_m \left(\frac{m+2}{m}\right)^{\frac{m+2}{2}}}{M^{2/m}} |K_{\mathbf{X}_m}|^{\frac{1}{m}}, & m \geq 2 \end{cases}$$

where h^*_m is the least normalized moment of inertia for a m-dimensional polytope. Results for normalized $\sigma^2_{\mathbf{Z},m,M}$ for predictors of order $m = 2$ with various M are shown in Figure 2. Note that if too few few states are used, the performance can be worse than the unquantized first-order predictor.

5 Design and Numerical Results

From (1) and the results in Figures 1 and 2, it is suggested that higher-order Markov models can yield substantial gains in the performance characterization of communication systems operating over fading channels. However, these results do not suggest a particular design for the state space of the Markov models. In this section, three design techniques for second-order models are introduced. Whereas the results of the previous section were based on a quantized complex Gaussian random process, the Markov model design considers the quantization of the Rayleigh fading process. A Markov process is completely described by $\{S, \pi, \mathbf{P}\}$ where S is the state space, π is a vector of steady-state probabilities, and $\mathbf{P}$ is a matrix of transition probabilities.

5.1 Product Quantizer Approach

The product quantizer is a straightforward method to generate a second-order Markov model to approximate the Rayleigh fading channel characteristics. To generate the state space in the product quantizer design, a Lloyd-Max quantizer is first used to partition the space for a single outdated estimate. If a total of M states are available, then $\sqrt{M}$ states are used by the Lloyd-Max quantizer to generate a mapping. Once this partition is constructed, the M-state two-dimensional quantizer is generated by choosing all possible pairs of states from the Lloyd-Max quantizer. Once the appropriate state space is generated, it is straightforward to compute the steady state probabilities and the transition matrix.

5.2 Vector Quantizer Approach

Although the product quantizer leads to a simple higher-order model, it is not an optimal design as can be seen by the fact that regardless of how the channel conditions change, the state space S remains static. The correlated nature of the fading channel can be exploited by using an m-dimensional vector quantizer to partition the state space of the Markov model. Based on a training sequence with appropriate statistical properties, the Linde-Buzo-Gray (LBG) algorithm is used to generate S. After initializing a set of M centroids, the LBG algorithm finds the best partition S from a set of centroids by using a *Euclidean* distance metric to find the centroid closest to the sample points of the training sequence. The algorithm then computes new centroids and repeats the process to find the best quantization. The Markov model parameters, π and $\mathbf{P}$, are generated for S from a simulated fading process.

5.3 Weighted Vector Quantization

Motivated by the importance of prediction error in the criterion of Section 3, the third approach incorporates the MMSE prediction error in the determination of the quantization of the state space. Instead of using a Euclidean distance metric, the following weighted error criterion is used to generate the quantizer partitions:

$$\xi^2 = (1-\lambda)\left[\hat{C}(c_0, c_1 \ldots c_{m-1}) - \hat{C}(Y_0, Y_1 \ldots Y_{m-1})\right]^2 + \lambda \sum_{k=0}^{m-1} (c_k - Y_k)^2 \qquad (2)$$

for $0 \leq \lambda \leq 1$, where λ is a weighting factor, $(c_0, c_1 \ldots c_{m-1})$ is a point in the state space, $(Y_0, Y_1 \ldots Y_{m-1})$ denotes the centroid of a state partition in S, and $\hat{C}$ is the linear

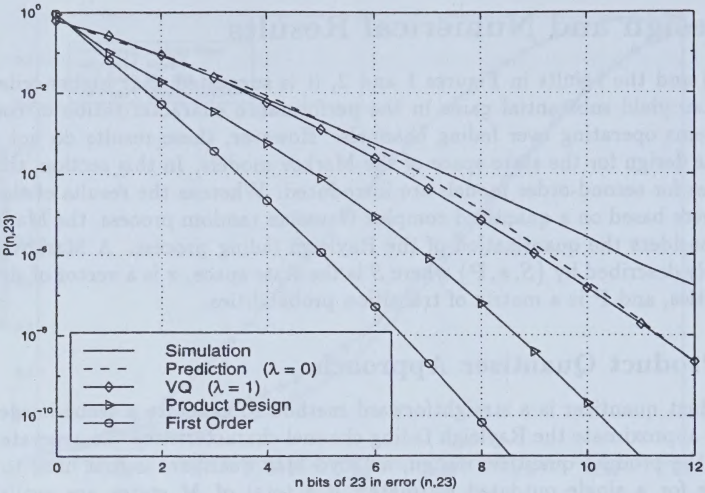


Figure 3: $P(n, 23)$ as a function of n when various channel models are used for $M = 16$, $E_s/N_0 = 10$ dB and $f_dT = 0.06$.

MMSE predictor based on m outdated fading estimates. The second term in the error criterion corresponds to the Euclidean distance metric. The first term is a prediction-error term that biases the quantization such that regions of constant prediction are quantized into the same partition of the state space. This change in the distortion criterion only affects how points are associated with centroids; the rest of the LBG algorithm remains unaltered. Hence, this weighted criterion attempts to incorporate the criterion of Section 3 while still preserving some of the process tracking capabilities of a direct vector quantization of the state space.

5.4 Error Recursion Results

Error recursions are often used to analyze the performance of coded communication systems in channels with memory [9]. Here, the recursions will be used to compare the the performance of a first-order Markov model where states are quantized by the Lloyd-Max algorithm, the second-order design techniques described above, and the simulated fading channel. Unless otherwise specified, each Markovian model employs the same number of states. Define $P(n, N)$ as the probability of n errors in N consecutive bits. For the sake of brevity, the analysis used to obtain $P(n, N)$ will not be described.

In Figure 3, $P(n, N)$ for the models considered is plotted as a function of n, the number of bits in error for $N = 23$. For the simulated process, no error events were statistically significant for $n \geq 12$. The numerical analysis of the error recursions shows that each of the second-order models is significantly more accurate than the first-order model in characterizing the performance of the system. Using the weighted criterion, the best performance occurs when $\lambda = 0$ – corresponding to the case when S is generated entirely based on the prediction criterion. However, the vector quantization approach still generates very accurate results. As expected, the unoptimized product approach performs the worst of the second-order approaches, although it is still markedly better than the first-order model. This suggests that simply moving from a first-order approach

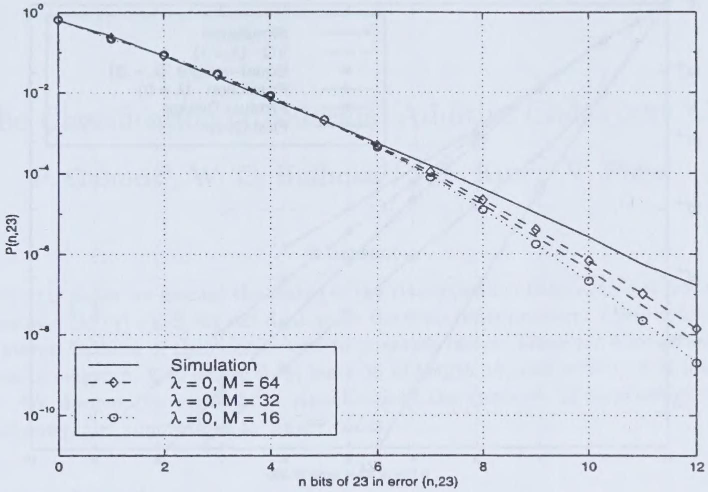


Figure 4: $P(n, 23)$ as a function of n for prediction-based Markov model when the state size M is varied for $E_s/N_0 = 10$ dB and $f_dT = 0.06$.

to a second-order model significantly improves the modeling of the channel.

Figure 4 considers the error performance for the prediction-based second-order model as a function of M, the number of states. The same channel parameters from the previous figure are employed. As expected, using more states in the Markov model results in performance that more closely matches the results from simulated fading. However, the likely error events match well for all of the models; the improvement in using more states is from the less likely error events. For practical applications, this suggests that only a moderate number of states is necessary for the second-order Markov chain to perform well.

In Figure 5, the error statistics for various Markov models are plotted for a channel with a higher rate of variation. While most of the results are consistent with Figure 3, it is interesting to note that the best performance of a second-order vector quantization approach occurs when $\lambda = 0$ or $\lambda = 0.5$ is used in (2); this corresponds to the the basic vector quantization case and a criterion which equally weights Euclidean distance and prediction error, respectively. This could be an indication that the best second-order construction does not depend only on (1). Generating the best higher-order model is still not well understood.

6 Conclusions

This work clearly demonstrates - both from theory and practice - the advantage of using higher-order Markov models in the performance characterization of wireless systems transmitting in correlated fading channels. Furthermore, it has motivated and demonstrated that a straightforward vector quantization approach to state space design of the higher-order models can be suboptimal. This motivates the continuing effort of establishing the correct criteria for the state space design of higher order Markov models for fading channels.

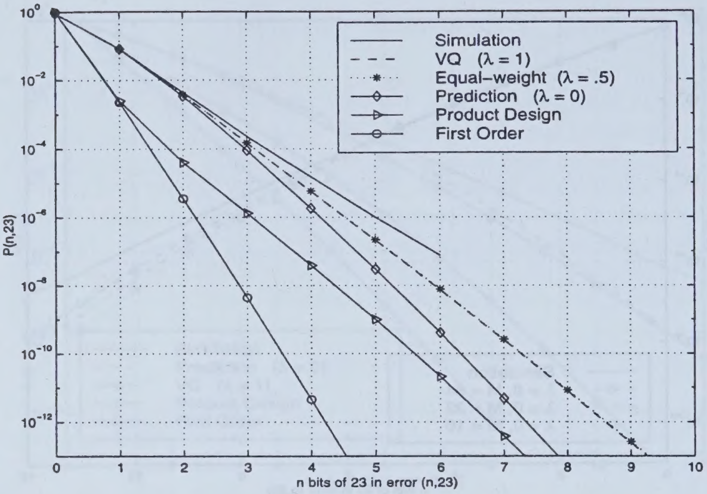


Figure 5: $P(n, 23)$ as a function of n when various channel models are used for $M = 64$, $E_s/N_0 = 18$ dB and $f_dT = 0.12$.

References

[1] R. McEliece and W. Stark, "Channels with Block Interference," *IEEE Trans. on Information Theory,* Vol. 30, pp. 44-53, January 1984.

[2] H. Wang and N. Moayeri, "Finite-State Markov Channel - A Useful Model for Radio Communication Channels," *IEEE Trans. Vehicular Technology,* Vol. 44, pp.163-171, February 1995.

[3] H. Wang and P. Chang, "On Verifying the First-Order Markovian Assumption for Rayleigh Fading Channel Model," *IEEE Trans. Vehicular Technology,* Vol. 45, pp. 353-357 May 1996.

[4] W. Turin and R. van Nobelen, "Hidden Markov Modeling of Flat Fading Channels," *IEEE Journal on Selected Areas in Communications,* Vol. 16, pp 1809-1817, December 1998.

[5] F. Gagnon and D. Haccoun, "Bounds on the Error Performance of Coding for Nonindependent Rician-Fading Channels," *IEEE Transactions on Communications,* Vol. 40, pp. 351-360, February 1992.

[6] G. Turin, "The Characteristic Function of Hermitian Quadratic Forms in Complex Normal Variables," *Biometrika,* Vol. 47, pp. 199-201, June 1960.

[7] T. Cover and J. Thomas, *Elements of Information Theory,* New York: Wiley, 1991.

[8] R. Gray and D. Neuhoff, "Quantization," *IEEE Transaction on Information Theory,* Vol. 44, pp. 2325-2383, October 1998.

[9] L. Kanal and A. Sastry, "Models for Channels with Memory and Their Applications to Error Control," *Proceedings IEEE,* Vol. 66, pp. 724-744, July 1978.

On the Classification of Extremal Additive Codes over $GF(4)$

P. Gaborit*, W. C. Huffman†, J.-L. Kim‡, V. Pless‡

Abstract

In this paper we discuss the status of the classification of the extremal additive codes over $GF(4)$ which are self-dual under the trace inner product. These codes are of interest because of their connection to quantum codes. There are 5 inequivalent codes of length 8, 8 of length 9, at least 56 of length 10, and exactly 1 of length 11. We discuss the methods of classification, the concepts of lengthening and shortening, and connections to binary codes.

1 Introduction

An *additive code* $\mathcal{C}$ *over* $GF(4)$ *of length* n is an additive subgroup of $GF(4)^n$. As $\mathcal{C}$ is a free $GF(2)$-module, it has size 2^k for some $0 \le k \le 2n$. We call $\mathcal{C}$ an $(n, 2^k)$ code. It has a basis, as a $GF(2)$-module, consisting of k basis vectors; a *generator matrix* of $\mathcal{C}$ will be a $k \times n$ matrix with entries in $GF(4)$ whose rows are a basis of $\mathcal{C}$. Interest in additive codes over $GF(4)$ has arisen because of their correspondence to quantum codes as described in [1]. There is a natural inner product arising from the trace map, on the additive codes induced by this correspondence. If we let $GF(4) = \{0, 1, \omega, \overline{\omega}\}$ where $\overline{\omega} = \omega^2 = 1 + \omega$, the *trace* map $\mathrm{Tr} : GF(4) \to GF(2)$ is given by

$$\mathrm{Tr}(x) = x + x^2.$$

In particular $\mathrm{Tr}(0) = \mathrm{Tr}(1) = 0$ and $\mathrm{Tr}(\omega) = \mathrm{Tr}(\overline{\omega}) = 1$. The *conjugate* of $x \in GF(4)$, denoted $\overline{x}$, is the image of x under the Frobenius automorphism; in other words, $\overline{0} = 0$, $\overline{1} = 1$, and $\overline{\overline{\omega}} = \omega$. We now define the *trace inner product* of two vectors $\mathbf{x} = x_1 x_2 \cdots x_n$ and $\mathbf{y} = y_1 y_2 \cdots y_n$ in $GF(4)^n$ to be

$$\mathbf{x} \star \mathbf{y} = \sum_{i=1}^{n} \mathrm{Tr}(x_i \overline{y_i}).$$

Example 1.1 Let $\mathcal{G}_6$ be the $[6, 3, 4]$ *hexacode* whose generator matrix as a linear $GF(4)$-code is

$$\begin{bmatrix} 1 & 0 & 0 & 1 & \omega & \omega \\ 0 & 1 & 0 & \omega & 1 & \omega \\ 0 & 0 & 1 & \omega & \omega & 1 \end{bmatrix}.$$

---

[*] LACO, Université de Limoges, France; e-mail: gaborit@math.uic.edu

[†] Department of Mathematical and Computer Sciences, Loyola University, Chicago, IL 60626, USA; e-mail: wch@math.luc.edu

[‡] Department of Mathematics, Statistics, and Computer Science, 322 SEO, University of Illinois–Chicago, 851 S. Morgan, Chicago, IL 60607-7045, USA; e-mail: jlkim@math.uic.edu, pless@math.uic.edu

This is also an additive code; thinking of $\mathcal{G}_6$ as an additive code, it has generator matrix

$$\begin{bmatrix} 1 & 0 & 0 & 1 & \omega & \omega \\ \omega & 0 & 0 & \omega & \overline{\omega} & \overline{\omega} \\ 0 & 1 & 0 & \omega & 1 & \omega \\ 0 & \omega & 0 & \overline{\omega} & \omega & \overline{\omega} \\ 0 & 0 & 1 & \omega & \omega & 1 \\ 0 & 0 & \omega & \overline{\omega} & \overline{\omega} & \omega \end{bmatrix}.$$

If $\mathcal{C}$ is an additive code, its *dual*, denoted $\mathcal{C}^\perp$, is the additive code $\{\mathbf{x} \in GF(4)^n \mid \mathbf{x} \star \mathbf{c} = 0 \text{ for all } \mathbf{c} \in \mathcal{C}\}$. If $\mathcal{C}$ is an $(n, 2^k)$ code, then $\mathcal{C}^\perp$ is an $(n, 2^{2n-k})$ code. As usual, $\mathcal{C}$ is *self-orthogonal* if $\mathcal{C} \subseteq \mathcal{C}^\perp$ and *self-dual* if $\mathcal{C} = \mathcal{C}^\perp$. In particular, if $\mathcal{C}$ is self-dual, $\mathcal{C}$ is an $(n, 2^n)$ code. The code $\mathcal{G}_6$ in Example 1.1 is self-dual as an additive code. (Any $GF(4)$-linear code that is self-orthogonal under the Hermitian inner product is a self-orthogonal additive code under the trace inner product.)

As usual, the *weight* $\operatorname{wt}(\mathbf{c})$ of $\mathbf{c} \in \mathcal{C}$ is the number of nonzero components of $\mathbf{c}$. The minimum weight d of $\mathcal{C}$ is the smallest weight of any nonzero codeword in $\mathcal{C}$. If $\mathcal{C}$ is an $(n, 2^k)$ additive code of minimum weight d, $\mathcal{C}$ is called an $(n, 2^k, d)$ code. $\mathcal{C}$ is *Type II* if $\mathcal{C}$ is self-dual and all codewords have even weight; as we see later Type II codes of length n exist only if n is even. If $\mathcal{C}$ is self-dual but some codeword has odd weight (in which case the code cannot be $GF(4)$-linear), the code is *Type I*. There is a bound on the minimum weight of an additive self-dual code [7, Theorem 33]. If d_I, d_{II} is the minimum distance of an additive self-dual Type I, Type II code, respectively, of length $n > 1$, then

$$d_I \leq \begin{cases} 2\left\lfloor \frac{n}{6} \right\rfloor + 1 & \text{if } n \equiv 0 \pmod 6 \\ 2\left\lfloor \frac{n}{6} \right\rfloor + 3 & \text{if } n \equiv 5 \pmod 6 \\ 2\left\lfloor \frac{n}{6} \right\rfloor + 2 & \text{otherwise} \end{cases} \tag{1}$$

$$d_{II} \leq 2\left\lfloor \frac{n}{6} \right\rfloor + 2. \tag{2}$$

A code that meets the appropriate bound is called *extremal.*

We say that two additive codes $\mathcal{C}_1$ and $\mathcal{C}_2$ are *equivalent* provided there is a map sending the codewords of $\mathcal{C}_1$ onto the codewords of $\mathcal{C}_2$ where the map consists of a permutation of coordinates followed by a scaling of coordinates by elements of $GF(4)$ followed by conjugation of some of the coordinates. In terms of generator matrices, if G_1 is a generator matrix of $\mathcal{C}_1$, then there is a generator matrix of $\mathcal{C}_2$ obtained from G_1 by permuting columns, scaling columns, and then conjugating some of the resulting columns. Notice that permuting coordinates, scaling coordinates, and conjugating some coordinates of a self-orthogonal (or self-dual) code does not change self-orthogonality (or self-duality). The *automorphism group* of $\mathcal{C}$, denoted $\operatorname{Aut}(\mathcal{C})$, consists of all maps which permute coordinates, scale coordinates, and conjugate coordinates that send codewords of $\mathcal{C}$ to codewords of $\mathcal{C}$.

The purpose of this paper is to examine the classification of the extremal Type I and II additive codes.

2 Results

We summarize what is currently known about extremal (or optimal) Type I and II codes in Table 1. In the table, codes have length 'n'; 'd_I' is the largest minimum weight for

n	d_I	num$_I$	d_{II}	num$_{II}$	n	d_I	num$_I$	d_{II}	num$_{II}$
2	1^O	1	2^E	1	10	4^E	≥ 51	4^E	≥ 5
3	2^E	1	–	–	11	5^E	1	–	–
4	2^E	1	2^E	1	12	5^E	≥ 7	6^E	1
5	3^E	1	–	–	13	5^O	≥ 5	–	–
6	3^E	1	4^E	1	14	6^E	?	6^E	≥ 1
7	3^O	3	–	–	15	6^E	≥ 2	–	–
8	4^E	2	4^E	3	16	6^E	≥ 3	6^E	≥ 4
9	4^E	8	–	–					

Table 1: Type I and II Codes of Length $2 \leq n \leq 16$

which a Type I code is known to exist while 'd_{II}' is the largest minimum weight for which a Type II code is known to exist. The superscript 'E' indicates that the code is extremal; the superscript 'O' indicates the code is not extremal but no code of the given type can exist with a larger minimum weight (and so the code is *optimal*). The number of inequivalent Type I and II codes of the given minimum weight is listed under 'num$_I$' and 'num$_{II}$', respectively; when the number in the column is exact (without $\geq$), the classification of those codes is complete.

The numbers for $2 \leq n \leq 7$ come from the work of Höhn [5]. Höhn also found that there are exactly 3 inequivalent extremal Type II codes of length 8; we found that there are exactly 2 inequivalent extremal Type I codes of length 8. We also found the entries for $n = 9$, 10, and 11. The unique extremal Type II code of length 12 is called the *dodecacode* [1]; as we will see, using the fact that the extremal self-dual code of length 11 is known to be unique, it is easy to show that the extremal Type II code of length 12 is also unique. It is known [7] that there is no extremal length 13 codes. We found that there are at least 5 optimal self-dual codes of length 13 and at least 3 extremal Type I codes of length 16; there are exactly 4 extremal Hermitian self-dual $GF(4)$-linear codes of length 16 [3] showing that there are at least 4 extremal Type II codes of length 16. The entries for length $n = 14$ comes from [7].

3 Techniques

In this section, we describe the techniques used to produce some of the results in the previous section.

3.1 The Balance Principle

We notice that a result found in [6] applies to additive self-dual codes with minor modifications. This is sometimes called the Balance Principle.

Theorem 3.1 *Let C be a self-dual additive $(n, 2^n)$ code. Let $n = n_1 + n_2$. We can choose a generator matrix G for C of the following form:*

$$G = \begin{bmatrix} A & O \\ O & B \\ D & E \end{bmatrix}. \tag{3}$$

Here A is a $k_1 \times n_1$ matrix where $[A \quad O]$ generates the largest subcode of $\mathcal{C}$ which is zero on the right-most n_2 coordinates. Similarly B is a $k_2 \times n_2$ matrix where $[O \quad B]$ generates the largest subcode of $\mathcal{C}$ which is zero on the left-most n_1 coordinates. (O is the appropriate size zero matrix.) Let $\mathcal{A}$ be the additive code of length n_1 generated by A, $\mathcal{A}_D$ the additive code of length n_1 generated by the rows of A and D, $\mathcal{B}$ the additive code of length n_2 generated by B, and $\mathcal{B}_E$ the additive code of length n_2 generated by the rows of B and E. The following hold.

(i) $n - k_2 = 2n_1 - k_1$ *and* $n - k_1 = 2n_2 - k_2$.

(ii) $\mathcal{A}^\perp = \mathcal{A}_D$ *and* $\mathcal{B}^\perp = \mathcal{B}_E$.

We apply the Balance Principle in the following manner. Suppose we wish to construct a self-dual additive $(n, 2^n, d)$ code $\mathcal{C}$. By rearranging coordinates, we may assume that there is a minimum weight codeword whose nonzero components are the left-most $n_1 = d$. Then by scaling the coordinates appropriately, we may assume that the first row of A is all 1's. There is at most one more row of A and that row can have only ω's and $\overline{\omega}$'s. By conjugation of columns, we may assume that row has all ω's; as the two rows must be orthogonal under the trace inner product, d must be even. Thus we can always choose A to be either

$$(a) \quad A = \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \end{bmatrix} \quad \text{or} \quad (b) \quad A = \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ \omega & \omega & \omega & \cdots & \omega \end{bmatrix} \tag{4}$$

with (b) occurring only when d is even. Thus we have $k_1 = 1$ or $k_1 = 2$. This determines the value for k_2.

Example 3.2 If $\mathcal{C}$ is a self-dual additive $(8, 2^8, 4)$ code, then the Balance Principle shows that a generator matrix G for $\mathcal{C}$ can be assumed to be of the form

$$\left[\begin{array}{cccc|cccc}
1 & 1 & 1 & 1 & & & & \\ \hline
 & & & & 1 & 1 & 1 & 1 \\ \hline
1 & 0 & 0 & 0 & & & & \\
0 & 1 & 0 & 0 & & & & \\
0 & 0 & 1 & 0 & & E & & \\
\omega & \omega & 0 & 0 & & & & \\
\omega & 0 & \omega & 0 & & & & \\
\omega & 0 & 0 & \omega & & & &
\end{array}\right] \quad \text{or} \quad \left[\begin{array}{cccc|cccc}
1 & 1 & 1 & 1 & & & & \\
\omega & \omega & \omega & \omega & & & & \\ \hline
 & & & & 1 & 1 & 1 & 1 \\
 & & & & \omega & \omega & \omega & \omega \\ \hline
1 & 1 & 0 & 0 & & & & \\
\omega & \omega & 0 & 0 & & E & & \\
1 & 0 & 1 & 0 & & & & \\
\omega & 0 & \omega & 0 & & & &
\end{array}\right].$$

Using these two forms, self-duality, the minimum weight, and equivalence, MAGMA was used to fill in the possible entries for E and thereby produce exactly 5 inequivalent self-dual additive $(8, 2^8, 4)$ codes, 3 being of Type II.

Example 3.3 Suppose we want to produce a self-dual additive $(9, 2^9, 4)$ code $\mathcal{C}$. We let a generator matrix for $\mathcal{C}$ have form (3). If A has form (4)(a), up to equivalence there are 3 choices for the 2×5 matrix B; if A has form (4)(b), up to equivalence there is only 1 choice for the 3×5 matrix B. Given A, D can be explicitly written down and E can be determined by MAGMA. These lead to exactly 8 inequivalent codes.

Example 3.4 Suppose we want to produce a self-dual additive $(11, 2^{11}, 5)$ code $\mathcal{C}$. A generator matrix for $\mathcal{C}$ has form (3) where this time A has form (4)(a) only and D can explicitly be written down. Up to equivalence there are 2 choices for the 2×6 matrix B. Using MAGMA E can be determined. This leads to exactly 1 code up to equivalence.

3.2 Lengthening and Shortening

If $\mathcal{C}$ is a self-dual additive $(n, 2^n, d)$ code, we can obtain a code of length $n-1$ by a process we call *shortening.* We use this term even though it differs from the usual meaning as our notion can be regarded as a generalization. Let G be a generator matrix of $\mathcal{C}$. As $\mathcal{C}$ is self-dual, no column of G is all 0's. Choose any column of G, say the i^{th} one. The entries in column i can be any of 0, 1, ω, or $\overline{\omega}$. By row reducing G (to obtain another generator matrix we call G again), we can make all the entries in column i equal to 0 except for one or two entries; if two they would be two of the three values 1, ω, or $\overline{\omega}$. The *shortened code of* $\mathcal{C}$ *on coordinate* i, denoted $\mathcal{C}'$ is the code with generator matrix G' obtained from G by eliminating one row of G with a nonzero entry in column i and then eliminating column i. If there is only one nonzero entry in column i of G, then $\mathcal{C}'$ is $\mathcal{C}$ shortened in the usual sense. Clearly $\mathcal{C}'$ is a self-dual additive $(n-1, 2^{n-1}, d')$ code with $d' \geq d-1$.

Example 3.5 Let

$$G = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ \omega & \omega & \omega \end{bmatrix}.$$

We obtain two inequivalent codes by shortening on the first column. They have the following generator matrices:

$$\begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix} \quad \text{or} \quad \begin{bmatrix} 1 & 1 \\ \omega & \omega \end{bmatrix}.$$

We would get the same codes shortening on the second column. To shorten on the third column, we would replace G by

$$\begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 0 \\ \omega & \omega & \omega \end{bmatrix}.$$

Shortening yields the same codes as before.

Example 3.6 If

$$G = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix},$$

then no matter which column we shorten on, we obtain a shortened code with generator matrix

$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix},$$

which is the usual shortened code. It also is the same as the first of the shortened codes in Example 3.5.

From Examples 3.5 and 3.6, we see that inequivalent codes can have equivalent shortened codes.

We can reverse shortening by *lengthening* a self-dual additive $(n-1, 2^{n-1}, d')$ code $\mathcal{C}'$ to obtain a self-dual additive $(n, 2^n, d)$ code $\mathcal{C}$. To do this, take a generator matrix G' of $\mathcal{C}'$ and adjoin an arbitrary n^{th} row $\mathbf{x}$ at the bottom. Then add an n^{th} column on the right according to the following scheme: First place a 1 in the n^{th} row. Second place a

0 or ω in each of the first $n-1$ rows so that each of the new length n rows is orthogonal to the length n bottom row. (Note that if we begin with two orthogonal rows of length $n-1$ and want to adjoin a nonzero element to one of the rows, we must adjoin the same nonzero element to the other row if we wish to create two orthogonal rows of length n.) The resulting matrix G, which we denote by

$$G = \left[\begin{array}{c|c} & 0 \\ G' & \text{or} \\ & \omega \\ \hline \text{x} & 1 \end{array}\right], \tag{5}$$

clearly generates a self-dual additive $(n, 2^n, d)$ code with $d \leq d' + 1$. If the new row of length $n-1$ added to the bottom of G' is already in C', the lengthened code is a direct sum of C' and the $(1, 2, 1)$ code generated by $[1]$.

Example 3.7 Let

$$G' = \begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix}.$$

If we adjoin the row $[\omega\ \omega]$, then

$$G = \begin{bmatrix} 0 & 1 & \omega \\ 1 & 1 & 0 \\ \omega & \omega & 1 \end{bmatrix}.$$

Note that by multiplying column 3 by $\overline{\omega}$ and then conjugating column 3, we obtain a generator matrix for the $(3, 2^3)$ code of Example 3.5.

Example 3.8 Let

$$G' = \begin{bmatrix} 1 & 1 \\ \omega & \omega \end{bmatrix}.$$

If we adjoin the row $[1\ 0]$, then

$$G = \begin{bmatrix} 1 & 1 & 0 \\ \omega & \omega & \omega \\ 1 & 0 & 1 \end{bmatrix},$$

which is the code $(3, 2^3)$ of Example 3.5.

Examples 3.5 and 3.6 show that equivalent codes can be obtained by lengthening inequivalent codes.

We want to expand the concept of lengthening. Instead of (5) we could use the following generator matrices to lengthen:

$$\left[\begin{array}{c|c} & 0 \\ G' & \text{or} \\ & \overline{\omega} \\ \hline \text{x} & 1 \end{array}\right], \quad \left[\begin{array}{c|c} & 0 \\ G' & \text{or} \\ & 1 \\ \hline \text{x} & \omega \end{array}\right], \quad \left[\begin{array}{c|c} & 0 \\ G' & \text{or} \\ & \overline{\omega} \\ \hline \text{x} & \omega \end{array}\right], \quad \left[\begin{array}{c|c} & 0 \\ G' & \text{or} \\ & 1 \\ \hline \text{x} & \overline{\omega} \end{array}\right], \text{ or } \left[\begin{array}{c|c} & 0 \\ G' & \text{or} \\ & \omega \\ \hline \text{x} & \overline{\omega} \end{array}\right]. \tag{6}$$

However by scaling and/or conjugating the last column of the matrices in (6), we obtain the matrix (5) giving the following result.

Lemma 3.9 *The codes generated by any of the matrices in* (5) *or* (6) *are equivalent.*

As a result of this lemma we expand our definition of lengthening to allow us to lengthen using any of the six matrices in (5) and (6). Up to equivalence lengthening and shortening are inverse operations. Thus we have the following theorem.

Theorem 3.10 *Every self-dual additive* $(n-1, 2^{n-1})$ *code is shortened from a self-dual additive* $(n, 2^n)$ *code, and every self-dual additive* $(n, 2^n)$ *code is lengthened from a self-dual additive* $(n-1, 2^{n-1})$ *code.*

Example 3.11 Every self-dual additive $(13, 2^{13}, 5)$ code comes from lengthening a self-dual additive $(12, 2^{12}, d)$ code with $d \geq 4$. Lengthening the dodecacode we obtained 4 inequivalent $(13, 2^{13}, 5)$ codes; not all possibilities were attempted. One can also shorten a length 14 code to obtain a length 13 code; shortening the length 14 extended quadratic residue code q_{14} leads to a $(13, 2^{13}, 5)$ code as indicated in [7]. The latter is not equivalent to the four codes found from lengthening.

Example 3.12 By lengthening q_{14}, we found one self-dual additive $(15, 2^{15}, 6)$ code inequivalent to the known $(15, 2^{15}, 6)$ cyclic code c_{15} found in [7].

Example 3.13 Every self-dual additive $(16, 2^{16}, 6)$ code comes from lengthening a self-dual additive $(15, 2^{15}, d)$ code with $d \geq 5$. Lengthening c_{15} yielded 3 inequivalent $(16, 2^{16}, 6)$ Type I codes; again not all possibilities were attempted.

In [5], Höhn shows that a Type II code of length n exists only if n is even. If n is even, he defines (analogously to the situation of doubly-even binary codes of length $8r$ and their length $8r-2$ children [2]) a child of a Type II length n code. Beginning with a self-dual additive code of length $n-1$, he also shows how to construct a parent, which will be of Type II. The length $n-1$ children are, in our terminology, shortenings of an even length n code. The parent of a length $n-1$ code $\mathcal{C}'$ is obtained by taking the lengthening vector $\mathbf{x}$ to be a vector not in $\mathcal{C}'$ but orthogonal to the subcode $\mathcal{C}_0$ of $\mathcal{C}'$ consisting of all its even weight vectors. The vector $\mathbf{x}$ must have odd weight; otherwise the code $\mathcal{C}_0$ and vector $\mathbf{x}$ generate an additive Type II code of odd length, a contradiction.

Example 3.14 Let

$$G = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ \omega & \omega & \omega & \omega \end{bmatrix}.$$

Then G generates a Type II code $\mathcal{C}$ of length 4. By shortening we obtain inequivalent children $\mathcal{C}_1$ and $\mathcal{C}_2$ with generator matrices

$$G_1 = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix} \quad \text{and} \quad G_2 = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ \omega & \omega & \omega \end{bmatrix}.$$

Rearrange the rows of G_1 to obtain

$$G_1' = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 0 & 1 \end{bmatrix}$$

so that the first two rows of G_1' generate the even weight subcode $\mathcal{C}_0$ of $\mathcal{C}_1$. Lengthening using the second form in (6) with $\mathbf{x} = [\omega\ \omega\ \omega]$ gives $\mathcal{C}$ with generator matrix

$$\begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ \omega & \omega & \omega & \omega \end{bmatrix}.$$

Lengthening $\mathcal{C}_2$ using form (5) with $\mathbf{x} = [1\ 0\ 0]$ gives $\mathcal{C}$ with generator matrix

$$\begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ \omega & \omega & \omega & \omega \\ 1 & 0 & 0 & 1 \end{bmatrix}.$$

Theorem 3.15 *Let $\mathcal{C}'$ be a Type I code of odd length $n-1$. Then there is a unique Type II code $\mathcal{C}$ of length n obtained by lengthening $\mathcal{C}'$.*

Proof: There is a unique $(n-1, 2^{n-2})$ subcode $\mathcal{C}_0$ of $\mathcal{C}'$ consisting of even weight vectors. Let G' be a generator matrix of $\mathcal{C}'$ with the first $n-2$ rows a basis of $\mathcal{C}_0$; the bottom row has odd weight. If we lengthen using (5) and we adjoin 0 to each of the first $n-2$ rows and ω to the last row, then we must choose the lengthening vector $\mathbf{x}$ to be in $\mathcal{C}_0^\perp$; to avoid a code of minimum weight 1, $\mathbf{x} \notin \mathcal{C}'$. Furthermore two choices of $\mathbf{x}$ in the same coset of $\mathcal{C}'$ give the same code $\mathcal{C}$. As there is only one coset of $\mathcal{C}'$ in $\mathcal{C}_0^\perp$ not equal to $\mathcal{C}'$, we have a unique lengthening to a Type II code. □

The code $\mathcal{C}$ in Theorem 3.15 is the parent of $\mathcal{C}'$. Thus the theorem states that an odd length Type I code has a unique Type II extension and that extension is the parent.

Example 3.16 By Theorem 3.10, (1), and (2), every self-dual additive $(10, 2^{10}, 4)$ code can be obtained from a self-dual additive $(9, 2^9, d)$ code with $d = 3$ or 4. We lengthened some of the 8 $(9, 2^9, 4)$ codes obtained in Example 3.3 and quickly found 51 Type I codes. Five of the 8 parents of the $(9, 2^9, 4)$ codes were inequivalent $(10, 2^{10}, 4)$ Type II codes; the other three parents had minimum weight 2. It seems to be of little interest to try to enumerate all self-dual additive $(10, 2^{10}, 4)$ codes as the number appears to be very high.

While the Type II extension of a Type I code of odd length is unique, this example shows that there may be many more Type I extensions. If we wish to extend Type II codes, we may get many extensions, as Example 3.11 illustrates. The following result was stated in [1] without proof.

Corollary 3.17 *The dodecacode is the unique $(12, 2^{12}, 6)$ Type II code.*

Proof: A $(12, 2^{12}, 6)$ Type II code must be a parent of a self-dual additive $(11, 2^{11}, d)$ code $\mathcal{C}'$ with $d \geq 5$. This code, being of odd length, must be Type I; by (1), $d = 5$. By Example 3.4, $\mathcal{C}'$ is unique, and so there is a unique $(12, 2^{12}, 6)$ Type II code by Theorem 3.15. □

4 Connections with Binary Codes

Höhn [5] presents two constructions that begin with a self-dual additive $(n, 2^n)$ code and produce a self-dual binary code of length $4n$. We describe these two (Constructions A and B) and add a third (C). If $\mathcal{C}$ is a self-dual additive $(n, 2^n)$ code, let $\widehat{\mathcal{C}}$ be the binary linear $[4n, n]$ code obtained from $\mathcal{C}$ by replacing each $GF(4)$ component by a 4-tuple in $GF(2)^4$ as follows: $0 \rightarrow 0000$, $1 \rightarrow 0110$, $\omega \rightarrow 1010$, $\overline{\omega} \rightarrow 1100$. Let d_4 be the $[4,1]$ binary linear code $\{0000, 1111\}$. Finally let $(d_4^n)_0$ be the $[4n, n-1]$ binary linear code consisting of all codewords of weights divisible by 8 from the $[4n, n]$ code d_4^n.

Construction A: $\rho_A(\mathcal{C}) = \widehat{\mathcal{C}} + d_4^n$

Construction B: Assume that n is even. $\rho_B(\mathcal{C}) = \widehat{\mathcal{C}} + (d_4^n)_0 + e_B$ where e_B is the $[4n, 1]$ code generated by

$$\begin{cases} 0001\ 0001\ \cdots\ 0001 & \text{if } n \equiv 0 \pmod 4 \\ 0001\ 0001\ \cdots\ 1110 & \text{if } n \equiv 2 \pmod 4 \end{cases}.$$

Construction C: Assume that n is even. $\rho_C(\mathcal{C}) = \widehat{\mathcal{C}} + (d_4^n)_0 + e_C$ where e_C is the $[4n, 1]$ code generated by

$$\begin{cases} 0001\ 0001\ \cdots\ 1110 & \text{if } n \equiv 0 \pmod 4 \\ 0001\ 0001\ \cdots\ 0001 & \text{if } n \equiv 2 \pmod 4 \end{cases}.$$

The next result is a straightforward extension of Lemma 2 of [5].

Theorem 4.1 *Let $\mathcal{C}$ be a self-dual additive code. The following hold.*

(i) *$\widehat{\mathcal{C}}$ is a self-orthogonal code.*

(ii) *$\rho_A(\mathcal{C})$, $\rho_B(\mathcal{C})$, and $\rho_C(\mathcal{C})$ are self-dual binary codes.*

(iii) *If $\mathcal{C}$ is Type II, $\rho_A(\mathcal{C})$ and $\rho_B(\mathcal{C})$ are doubly-even whereas if $\mathcal{C}$ is Type I, $\rho_A(\mathcal{C})$ and $\rho_B(\mathcal{C})$ are singly-even.*

(iv) *$\rho_C(\mathcal{C})$ is singly-even.*

We give a proof of the following, which we have alluded to throughout the paper and was first proved in [5] in a more complicated manner.

Theorem 4.2 *A self-dual additive Type II code of odd length n cannot exist.*

Proof: If $\rho_A(\mathcal{C})$ is a binary self-dual doubly-even code of length $4n$, then n is even as $4n$ must be a multiple of 8. The result follows by Theorem 4.1(iii). □

Example 4.3 Let $\mathcal{G}_6$ be the $(6, 2^6, 4)$ hexacode given in Example 1.1. We verified that $\rho_A(\mathcal{G}_6)$ is the $[24, 12, 8]$ binary Golay code and $\rho_C(\mathcal{G}_6)$ is the $[24, 12, 6]$ *odd Golay code*. It is interesting to note that beginning with the unique extremal self-dual additive code of length 6, we obtain the unique extremal self-dual doubly-even and singly-even binary codes of length 24 using constructions A and C. Note also by Table 2 that $\rho_C(\mathcal{G}_6)$ has automorphism group of size 2^6 times that of $\mathcal{G}_6$.

$\mathcal{C}$	$\lvert\mathrm{Aut}(\mathcal{C})\rvert$	$\lvert\mathrm{Aut}(\rho_B(\mathcal{C}))\rvert$	$\lvert\mathrm{Aut}(\rho_C(\mathcal{C}))\rvert$
$\mathcal{G}_6$	$2^4 \cdot 3^3 \cdot 5$	$2^{10} \cdot 3^3 \cdot 5 \cdot 7 \cdot 11 \cdot 23$	$2^{10} \cdot 3^3 \cdot 5$
$\mathcal{C}_1$	$2^7 \cdot 3^2$	$2^{15} \cdot 3^2 \cdot 5 \cdot 7$	$2^{15} \cdot 3^2$
$\mathcal{C}_2$	$2^4 \cdot 3 \cdot 7$	$2^{12} \cdot 3 \cdot 7$	$2^{12} \cdot 3 \cdot 7$
$\mathcal{C}_3$	$2^7 \cdot 3^2 \cdot 7$	$2^{15} \cdot 3^2 \cdot 5 \cdot 7 \cdot 31$	$2^{15} \cdot 3^2 \cdot 7$

Table 2: Some Automorphism Group Orders

Example 4.4 By Example 3.2 there are exactly three $(8, 2^8, 4)$ Type II codes $\mathcal{C}_i$ with generator matrices G_i where

$$G_1 = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 & 1 & \omega & \omega & 0 \\ 0 & 1 & 0 & 0 & 1 & \overline{\omega} & \overline{\omega} & 0 \\ 0 & 0 & 1 & 0 & \overline{\omega} & 1 & 0 & \omega \\ \omega & \omega & 0 & 0 & \omega & \omega & \omega & \omega \\ \omega & 0 & \omega & 0 & 1 & 1 & \omega & \omega \\ \omega & 0 & 0 & \omega & \overline{\omega} & \overline{\omega} & 1 & 1 \end{bmatrix}, \quad G_2 = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 & 1 & \omega & \omega & 0 \\ 0 & 1 & 0 & 0 & 1 & \omega & 0 & \omega \\ 0 & 0 & 1 & 0 & \overline{\omega} & 1 & 0 & \omega \\ \omega & \omega & 0 & 0 & \omega & 1 & \overline{\omega} & 1 \\ \omega & 0 & \omega & 0 & 1 & 0 & 1 & 0 \\ \omega & 0 & 0 & \omega & \overline{\omega} & \omega & \omega & \overline{\omega} \end{bmatrix}, \text{ and } G_3 = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ \omega & \omega & \omega & \omega & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & \omega & \omega & \omega & \omega \\ 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ \omega & \omega & 0 & 0 & \omega & \omega & 0 & 0 \\ 1 & 0 & 1 & 0 & \omega & \overline{\omega} & \omega & \overline{\omega} \\ \omega & 0 & \omega & 0 & \overline{\omega} & 1 & \overline{\omega} & 1 \end{bmatrix}.$$

Applying Construction B to each of these codes produces three of the five inequivalent self-dual doubly-even binary $[32, 16, 8]$ codes [2]. The codes $\rho_B(\mathcal{C}_i)$ are respectively C83 (or $2g_{16}$), C84 (or $8f_4$), and C82 (or r_{32}) in the notation of [2]. Notice that the extended quadratic residue code is not in this list but the second order Reed–Muller code is. Applying Construction C to each produces all three of the inequivalent self-dual singly-even binary $[32, 16, 8]$ codes [4]. Analogous to Example 4.3, $\rho_C(\mathcal{C}_i)$ has automorphism group of size 2^8 times that of $\mathcal{C}_i$ as shown in Table 2.

References

[1] A. R. Calderbank, E. M. Rains, P. W. Shor, and N. J. A. Sloane, "Quantum error correction via codes over $GF(4)$," *IEEE Trans. Inform. Theory* **IT-44** (1998), 1369–1387.

[2] J. H. Conway and V. Pless, "On the enumeration of self-dual codes," *J. Comb. Theory* **28A** (1980), 26–53.

[3] J. H. Conway, V. Pless, and N. J. A. Sloane, "Self-dual codes over $GF(3)$ and $GF(4)$ of length not exceeding 16," *IEEE Trans. Inform. Theory* **IT-25** (1979), 312–322.

[4] J. H. Conway and N. J. A. Sloane, "A new upper bound on the minimal distance of self-dual codes," *IEEE Trans. Inform. Theory* **IT-36** (1990), 1319–1333.

[5] G. Höhn, "Self-dual codes over the Kleinian four group," preprint, 1996.

[6] V. Pless, *Introduction to the Theory of Error-Correcting Codes*, 3rd ed. New York: J. Wiley & Sons, 1998.

[7] E. M. Rains and N. J. A. Sloane, "Self-dual codes," in *Handbook of Coding Theory*, ed. V. S. Pless and W. C. Huffman. Amsterdam: Elsevier, 1998, pp. 177–294.

Two Fast Algorithms in the Sudan Decoding Procedure†

*Gui-Liang Feng**

Abstract

In the Sudan decoding procedure, there are two stages: weighted curve fitting and factorization of bivariate polynomial. In this paper, we present a generalized Welch-Berlekamp algorithm for weighted curve fitting. The complexity of generalized Welch-Berlekamp algorithm is $O(sn^2)$, where s is the degree of y in $Q(x,y)$. We also present a fast factorization algorithm with $O(n^2)$ for multivariate polynomial $Q(x,y)$, when the degree s of y in $Q(x,y)$ is 2 or 3. Thus, the complexity of Sudan decoding procedure is $O(n^2)$, when $s = 2$ or 3.

I Introduction

We first introduce the problem of weighted curve fitting [1]. We define *weighted degree (Deg)* and *total order* ($<_T$) of polynomials in two variables x and y as follows:

- Let $Deg\ y = k-1$, then define $Deg\ x^{\mu} y^{\nu} \triangleq \mu + (k-1)\nu$.
- Define $x^{\mu} y^{\nu} <_T x^{\mu'} y^{\nu'}$ if and only if $Deg\ x^{\mu} y^{\nu} < Deg\ x^{\mu'} y^{\nu'}$ or if $Deg\ x^{\mu} y^{\nu} = Deg\ x^{\mu'} y^{\nu'}$ and $\nu < \nu'$. Sometimes, we use (μ,ν) to express the monomial $x^{\mu} y^{\nu}$.
- Let $Q(x,y)$ be a polynomial in variables x and y. Let $x^{\mu^*} y^{\nu^*}$ be the monomial in $Q(x,y)$ with the largest total order, then we call $x^{\mu^*} y^{\nu^*}$ as the leading term of $Q(x,y)$. We define the total order and weighted degree of $Q(x,y)$ by the total order and weighted degree of its leading term, respectively.

Given n points $\{ (x_1,y_1), (x_2,y_2), \cdots, (x_n,y_n) \}$ and parameter $k-1$, where $k < n$, n and k are positive integers, the problem is to find a polynomial $Q(x,y)$ with the smallest total order such that

$$Q(x_{\xi}, y_{\xi}) = 0 \qquad \text{for } \xi = 1, 2, \cdots, n. \tag{1.1}$$

This problem is seen to be a generalization of the rational interpolation problem considered by Welch and Berlekamp [2,3], which is finding a polynomial $Q(x,y)$ with smallest total order, where $Q(x,y) = D(x)y - N(x)$ and satisfies (1.1), to a weighted curve fitting problem. The rational interpolation problem is known to correspond to the problem of decoding Reed-Solomon codes up to half of the minimum distance. This problem can be solved by the well-known Welch-Berlekamp algorithm [2,3]. The complexity is $O(n^2)$. Some papers dealing with the interpolation problem and the Welch-Berlekamp algorithm have been published [4-7].

Recently, Sudan published a new decoding procedure for Reed-Solomon codes [1,8,9], which in principle can find all codewords within some Hamming distance τ to a received vector; and for low rate Reed-Solomon codes, τ can be greater than the packing radius $t \triangleq \lfloor \frac{d_{\min}-1}{2} \rfloor$ of the code. The Sudan decoding procedure has been described in some papers [10-14]. In the Sudan decoding procedure, the key

---

* Gui-Liang Feng is with the Center for Advanced Computer Studies, University of Southwestern Louisiana, Lafayette, LA. 70504, USA. This work was supported in part by NSF under Grant NCR-9804973.

step is to find $Q(x,y)$ satisfying (1.1) with the smallest total order, i.e., the weighted curve fitting problem. In this paper, the solution of the weighted curve fitting problem by a generalized Welch-Berlekamp algorithm and a fast factorization of bivariate polynomial with y-degree 2 or 3 is presented. The application of the two algorithms to decoding Reed-Solomon codes beyond the minimum distance using the Sudan decoding procedure is considered. The generalized Welch-Berlekamp algorithm has the complexity $O(sn^2)$, where s is the y-degree of $Q(x,y)$ and the fast factorization algorithm has complexity $O(n^2)$. Therefore, the complexity of Sudan decoding procedure is $O(n^2)$ for the Reed-Solomon codes over $GF(2^m)$ and the y-degree of $Q(x,y) \leq 3$.

In the next section, we first formulate a more general problem, which is finding the smallest initial set of linearly dependent columns in a matrix over an arbitrary field, which includes the weighted curve fitting problem as a special case. Then an alternative fundamental iterative algorithm (AFIA) is derived based on the fundamental iterative algorithm (FIA) in [15]. In Section III, we derive the generalized Welch-Berlekamp algorithm. In Section IV, we present a fast factorization of bivariate polynomial with y-degree ≤ 3 over $GF(2^m)$. Finally, we present our conclusions in Section V.

II. Alternative Fundamental Iterative Algorithm

We formulate the weighted curve fitting problem as follows: from the parameter $k-1$ and points $\{ (x_\xi, y_\xi) \mid \xi = 1, 2, ..., n \}$, we construct the following $n \times (n+1)$ matrix $\mathbf{M}$:

$$[\ 1, x, x^2, \ldots, x^{k-1}, y, x^k, xy, x^{k+1}, x^2y, \ldots, x^\mu y^\nu, \cdots\ ]\ . \tag{2.1}$$

Hereafter, if no confusion arises, $x^s y^t$ sometimes expresses a monomial and sometimes denotes the column $(\cdots, x_\xi^\mu y_\xi^\nu, \cdots)^T$ as well as polynomial $P(x,y)$, and $x^\mu y^\nu$ are arranged by the total order in the above matrix. As an example, let $k-1 = 2$ and $n = 16$, we have the 16×17 matrix,

$$[\ 1, x, x^2, y, x^3, xy, x^4, x^2y, y^2, x^5, x^3y\ xy^2, x^6, x^4y, x^2y^2, y^3, x^7\ ]\ .$$

To find the smallest total order polynomial $Q(x,y)$ satisfying (1.1) is then equivalent to determining the minimum l such that the first $(l+1)$ columns of $\mathbf{M}$ are linearly dependent, and in addition, finding the coefficients $f_{\mu,\nu}$ pertaining to this linear dependence, where $Q(x,y) = \sum f_{\mu,\nu} x^\mu y^\nu$.

With this viewpoint, the weighted curve fitting problem is then best treated as a special case of finding the smallest initial set of dependent columns in an $M \times N$ matrix over the field with rank less than N [15]. For the $\mathbf{M}$ matrix (2.1), that is to find smallest l and $c_1, \ldots, c_l$ such that

$$F(x_i, y_j) \triangleq x_i^{\mu_{l+1}} y_i^{\nu_{l+1}} + \sum_{j=1}^{l} c_j x_i^{\mu_j} y_i^{\nu_j} = 0 \quad \text{for } i = 1, 2, \ldots, M\ . \tag{2.2}$$

where, $x^{\mu_j} y^{\nu_j}$ denotes the column j for $j = 1, 2, \ldots, n+1$. For example, in (2.1), $(\mu_1, \nu_1) = (0,0)$, $(\mu_2, \nu_2) = (1,0)$, $(\mu_k, \nu_k) = (k-1, 0)$, $(\mu_{k+1}, \nu_{k+1}) = (0,1)$, and so on.

Let $F^{(i-1,\mu_j,\nu_j)}(x,y) = \sum_{\xi=1}^{j} f_\xi^{(i-1,\mu_j,\nu_j)} x^{\mu_\xi} y^{\nu_\xi}$, where $f_j^{(i-1,\mu_j,\nu_j)} = 1$, be defined as the polynomial with the property that

$$F^{(i-1,\mu_j,\nu_j)}(x_h, y_h) = 0 \ \text{ for } \ h \leq i-1\ . \tag{2.3}$$

Let

$$d_{i,j} \triangleq F^{(i-1,\mu_j,\nu_j)}(x_i, y_i)\ , \tag{2.4}$$

In addition, let $F^{(j)}(x,y)$ be the final polynomial of column j. Then the FIA [15] can be rewritten in the following alternative form, called the *Alternative Fundamental Iterative Algorithm (AFIA)*:

```
Step 1: Empty Tables $D$ and $C$, $1 \to j$, $1 \to r$, $x^{\mu_1} y^{\nu_1} \to F^{(1)}(x,y)$.
Step 2: Compute $d_{r,j} = F^{(j)}(x_r, y_r)$.
Step 3: If $d_{r,j} = 0$, then
    (a) if $r = M$, then $j-1 \to l$, $F^{(j)}(x,y) \to F(x,y)$, stop; we have a solution.
    (b) otherwise $r+1 \to r$, and return to Step 2.
Step 4: If $d_{r,j} \neq 0$, then
    (a) if there exists a $d_{r,u} \in D$ for some $1 \le u < j$, then
            $F^{(j)}(x,y) - \frac{d_{r,j}}{d_{r,u}} F^{(u)}(x,y) \to F^{(j)}(x,y)$
        and return to Step 3.
    (b) otherwise, $d_{r,j}$ is stored in $D$, $x^{\mu_{j+1}} y^{\nu_{j+1}} \to F^{(j+1)}(x,y)$ and $F^{(j)}(x,y)$ is stored in $C$, mark
        "×" at row $r$ and column $j$. If $j < N$, then $j+1 \to j$, $1 \to r$, and return to Step 2. Otherwise
        stop, as this problem has no solution ( when $N = n+1$ and $M = n$, there exists a unique solution).
```

Obviously, the complexity of *AFIA* is $O(n^3)$, because the algorithm is based on Gaussian elimination. On the other hand, Tables C and D should store at most $M-1$ final polynomials and discrepancies for each of the columns. In the next section, we will refine the algorithm for special matrix **M**. The complexity of the refined *AFIA* algorithm is $O(sn^2)$, where s is the maximal power of y in the **M**. The Tables C and D can be reduced significantly for larger n.

III. Generalized Welch-Berlekamp Algorithm

In this section, through a refinement of the AFIA on such (2.1) matrix, we derive a generalization of the Welch-Berlekamp algorithm for solving the weighted curve fitting problem.

Consider the matrix **M** in (2.1), let s denote max $\{ \nu_j \mid j = 1, 2, \ldots, n+1 \}$. From Step 4 of *AFIA*, it is known that the leading term in $F^{(j)}(x,y)$ is $x^{\mu_j} y^{\nu_j}$. In addition, at Step 4(b), $F^{(j+1)}(x,y) = x^{\mu_j} y^{\nu_j}$ is not necessary. In fact, $F^{(j+1)}(x,y)$ should be $x^{\mu_j} y^{\nu_j} + f(x,y)$, where $f(x,y) <_T x^{\mu_j} y^{\nu_j}$. Furthermore, it is better that there exists a maximal h such that $F^{(j+1)}(x_i, y_i) = 0$, for $i = 1, 2, \ldots, h$. Thus, the algorithm can reduce h loops from Step 3 to Step 4. We have the following lemma:

Lemma 3.1: Let u be the largest integer with $u \le j$, $\mu_u = \mu_{j+1} - 1$, and $\nu_u = \nu_{j+1}$. Let $F^{(u)}(x,y)$ be the final polynomial of column u with

$$F^{(u)}(x_i, y_i) = 0 \qquad \text{for} \quad i = 1, 2, \ldots, h$$

and

$$F^{(u)}(x_{h+1}, y_{h+1}) \neq 0 .$$

In *AFIA*, let

$$F^{(j+1)}(x,y) \triangleq (x - x_{h+1}) \times F^{(u)}(x,y), \tag{3.1}$$

we have that the leader term of $F^{(j+1)}(x,y)$ is $x^{\mu_{j+1}} y^{\nu_{j+1}}$ and

$$F^{(j+1)}(x_i,y_i) = 0 \quad \text{for } i = 1, 2, \ldots, h, h+1. \tag{3.2}$$

Proof: From (3.1), for $i = 1, 2, \ldots, h$, the second factor, i.e. $F^{(u)}(x_i,y_i)$, is zero; for $i = h+1$, the first factor $(x_{h+1} - x_{h+1})$ is zero, hence (3.2) is true. On the other hand, $F^{(n)}(x,y)$ has leading term $x^{\mu_u} y^{\nu_u}$, from (3.1) and $\mu_u = \mu_{j+1} - 1$, $F^{(j+1)}(x,y)$ has leading term $x^{\mu_u+1} y^{\nu_u} = x^{\mu_{j+1}} y^{\nu_{j+1}}$. □

Thus, the columns of matrix **M** can be partitioned into $s+1$ parts P_ν for $\nu = 0, 1, \ldots, s$ and each column of P_ν corresponds to $x^\mu y^\nu$. From Lemma 3.1 and *AFIA*, it is known that Tables C and D are needed to store one polynomial and its corresponding non-zero discrepancy. For convenience, let $Q^{(\nu)}(x,y)$ be the polynomial in x and y with leading term $x^\mu y^\nu$ for $\nu = 0, 1, 2, \ldots, s$. Let $q_i^{(\nu)} \triangleq Q^{(\nu)}(x_i,y_i)$ for $\nu = 0, 1, 2, \ldots, s$. Thus, applying the *AFIA* to **M** in (2.1), the *AFIA* can be improved to the following *Improved Fundamental Iterative Algorithm (IFIA)* for **M**:

```
Step 1: Empty Tables $D$ and $C$, $1 \to j$, $1 \to r$, $Q^{(\nu)}(x,y) = y^\nu$ for $\nu = 0, 1, \ldots, s$.
Step 2: Suppose $\nu = \nu_j$, then compute $q_r^{(\nu)} = F^{(\nu)}(x_r,y_r)$.
Step 3: If $q_r^{(\nu)} = 0$, then
    (a) if $r = M$, then $j-1 \to l$, $Q^{(\nu)}(x,y) \to Q(x,y)$, stop; we have a solution.
    (b) otherwise $r+1 \to r$, and return to Step 2.
Step 4: If $q_r^{(\nu)} \neq 0$, then
    (a) if there exists a $q_r^{(\kappa)} \in D$ for some $0 \le \kappa \le s$, then
            $Q^{(\nu)}(x,y) - \frac{q_r^{(\nu)}}{q_r^{(\kappa)}} \times Q^{(\kappa)}(x,y) \to Q^{(\nu)}(x,y)$
        and return to Step 3.
    (b) otherwise, $q_r^{(\nu)}$ and and $Q^{(\nu)}(x,y)$ are stored in Tables $D$ and $C$, respectively. If the column $j+1$ belongs the class $\sigma$, then $(x - x_k) \times Q^{(\sigma)}(x,y) \to Q^{(\sigma)}(x,y)$, where $q_k^{(\sigma)}$ and $Q^{(\sigma)}(x,y)$ are stored in Tables $D$ and $C$, respectively. If $j < N$, then $j+1 \to j$, $1 \to r$, and return to Step 2. Otherwise stop, as this problem has no solution ( when $N = n+1$ and $M = n$, there exists an unique solution).
```

Since for each $0 \le \nu \le s$, we will calculate $q_i^{(\nu)}$ from $i = 0$ to $i = n$, and at most n times repeat Step 4(a), and each calculation needs at most n scalar multiplications, the complexity of the *IFIA* for **M** is $O(sn^2)$. On the other hand, Tables C and D contain at most $s+1$ final polynomials and corresponding discrepancies, respectively. Thus, we have the following theorem:

Theorem 3.1: The time complexity and space complexity of the *IFIA* for **M** in (2.1) are $O(sn^2)$ and $O(n)$, respectively.

During implementation of the *IFIA* for **M** in (2.1), it can be seen that: (1) for each part P_ν of the columns, at Step 4(a) and Step 4(b) only, we can get new $Q^{(\nu)}(x,y)$ by $Q^{(\nu)}(x,y) - \frac{q_r^{(\nu)}}{q_r^{(\kappa)}} \times Q^{(\kappa)}(x,y) \to Q^{(\nu)}(x,y)$ or by $(x - x_k) \times Q^{(\sigma)}(x,y) \to Q^{(\sigma)}(x,y)$; (2) Tables C and D store only the latest final polynomial $Q^{(\nu)}(x,y)$ and its corresponding discrepancy, respectively, for $\nu = 0, 1, \ldots, s$. Thus, we can implement the *IFIA* for **M** from one row moving to the next row instead of column. Therefore, we have an equivalent algorithm, which is referred to the *generalized Welch-Berlekamp algorithm:*

Step 1: $Q_0^{(\nu)}(x,y) = y^\nu$ for $\nu = 0, 1, 2, \ldots, s, j = 0$.

Step 2: If $j < n$, then calculate $Q_j^{(\nu)}(x_{j+1}, y_{j+1}) = q_{j+1}^{(\nu)}$ and go to Step 3. Otherwise, stop and $Q_n^{(\tau)}(x,y)$ with smallest total order is the solution.

Step 3: There are two cases

- If $q_{j+1}^{(\nu)}$ are not all zero, for $\nu = 0, 1, 2, \ldots, s$, let $Q_j^{(\sigma)}(x,y)$ be the smallest total order with $q_{j+1}^{(\sigma)} \neq 0$. Then

$$Q_{j+1}^{(\sigma)}(x,y) \triangleq (x - x_{j+1}) \times Q_j^{(\sigma)}(x,y) \ ;$$

for $q_{j+1}^{(\nu)} \neq 0$ and $\nu \neq \sigma$,

$$Q_{j+1}^{(\nu)}(x,y) \triangleq Q_j^{(\nu)}(x,y) - \frac{q_{j+1}^{(\nu)}}{q_{j+1}^{(\sigma)}} \times Q_j^{(\sigma)}(x,y) \ ;$$

for $q_{j+1}^{(\nu)} = 0$,

$$Q_{j+1}^{(\nu)}(x,y) \triangleq Q_j^{(\nu)}(x,y) \ ;$$

$j + 1 \rightarrow j$, and go to Step 2.

- If $q_{j+1}^{(\nu)} = 0$, for $\nu = 0, 1, 2, \ldots, s$, then

$$Q_{j+1}^{(\nu)}(x,y) \triangleq Q_j^{(\nu)}(x,y) \ ;$$

$j + 1 \rightarrow j$, and go to Step 2.

When $s = 1$, the algorithm reduces to the well-known Welch-Berlekamp algorithm [1,2].

IV. Fast Factorization Algorithm

In this section, we consider only the Reed-Solomon codes over $GF(2^l)$, i.e, the factorization of multivariables over $GF(2^l)$. First let us consider the case of $s = 2$. Let

$$Q(x,y) = m(x)y^2 + d(x)y + n(x) = (y + C(x))(m(x)y + h(x)) \ . \tag{4.1}$$

Then we have

$$y^2 + d(x)y + m(x)n(x) = (y + m(x)C(x))(y + h(x)) \ . \tag{4.2}$$

Thus, the factorization of $Q(x,y)$ reduces to the factorization of $y^2 + d(x)y + m(x)n(x)$. Without loss of generality, let us consider the following factorization:

$$y^2 + D(x)y + N(x) = (y + c(x))(y + h(x)) \ , \tag{4.3}$$

where $D(x)$ and $N(x)$ are known and $c(x)$ is unknown. Let $deg\ m(x) = \nu$ and $p \triangleq k - 1 + \nu$. Thus, let $c(x) = \sum_{i=1}^{p} c_i x^i$, $D(x) = \sum_{i=1}^{p} d_i x^i$, and $N(x) = \sum_{i=1}^{2p} n_i x^i$.

From (4.3), we have

$$c(x)^2 + D(x)c(x) + N(x) = 0 \ . \tag{4.4}$$

Over $GF(2^l)$, we have

$$\sum_{i=0}^{p} c_i^2 x^{2i} + \Big(\sum_{i=0}^{p} c_i x^i\Big)\times\Big(\sum_{i=0}^{p} d_i x^i\Big) + \sum_{i=0}^{2p} n_i x^i = 0 \,. \tag{4.4'}$$

Assume that $d_0 = \cdots = d_{\sigma-1} = 0$ and $d_\sigma \neq 0$, for $\sigma \geq 0$. From (4.4'), we have: for $0 \leq t \leq 2p$

$$\begin{cases} (c_{\frac{t}{2}})^2 + \sum_{i=0}^{t-\sigma} c_i d_{t-i} + n_t = 0 & \text{for } t \textit{ is even} \\ \sum_{i=0}^{t-\sigma} c_i d_{t-i} + n_t = 0 & \text{for } t \textit{ is odd} \end{cases} , \tag{4.5}$$

where $c_t = d_t = 0$ for $t > p$.

Lemma 4.1: If $c_0, c_1, \ldots, c_{\lambda-1}$ are known and $\lambda > \sigma$, then c_λ can be calculated by (4.5).

Proof: Since $\lambda > \sigma$, we have $\lambda > \lfloor \frac{\lambda+\sigma}{2} \rfloor$. If $\lambda + \sigma$ is even, the from (4.5), we have

$$(c_{\lfloor \frac{\lambda+\sigma}{2} \rfloor})^2 + \sum_{i=0}^{\lambda-1} c_i d_{\lambda+\sigma-i} + c_\lambda d_\sigma + n_{\lambda+\sigma} = 0 \,.$$

Thus, we can uniquely find c_λ. Similarly, if $\lambda + \sigma$ is odd, we have

$$\sum_{i=0}^{\lambda-1} c_i d_{\lambda+\sigma-i} + c_\lambda d_\sigma + n_{\lambda+\sigma} = 0 \,.$$

c_λ can also be found. □

Now we are going to derive a fast algorithm to find $c(x)$. Let us consider two cases.

- *Case 1)*: $\sigma > 0$. From (4.5), we have

$$c_\mu^2 + n_{2\mu} = 0 \qquad \text{for } \mu = 0, 1, \ldots, \xi,$$

i.e., c_μ have unique solution, where ξ denotes $\lfloor \frac{\sigma}{2} \rfloor$.

Let $\xi+1 \leq \mu < \sigma$, then $\mu > 2\mu - \sigma$. Let $t = 2\mu$ and using (4.5), we have

$$(c_\mu)^2 + \sum_{i=0}^{2\mu-\sigma} c_i d_{2\mu-i} + n_{2\mu} = 0 \,.$$

In the above equation, c_i, for $i = 0, 1, \ldots, 2\mu-\sigma$, are all known and c_μ is unknown. Thus, c_μ has a unique solution. On the other hand, let $t = 2\sigma$ and using (4.5), we have

$$(c_\sigma)^2 + d_\sigma c_\sigma + \sum_{i=0}^{\sigma-1} c_i d_{2\sigma-i} + n_{2\sigma} = 0 \,.$$

In the above equation, Only c_σ is unknown, and $d_\sigma \neq 0$, c_σ has two solutions: $c_\sigma^{(1)}$ and $c_\sigma^{(2)}$. For each solution $c_0, \ldots, c_{\sigma-1}, c_\sigma^{(j)}$ for $j = 1, 2$, repeatedly using Lemma 4.1, we can find $c_i^{(j)}$ for $i = \sigma+1, \ldots, p$ and $j = 1, 2$. Thus, we have two solutions $c^{(1)}(x)$ and $c^{(2)}(x)$.

- *Case 2)*: $\sigma = 0$. From (4.5), we have

$$c_0^2 + c_0 d_0 + n_0 = 0 \,.$$

In this case c_0 has two solutions: $c_0^{(1)}$ and $c_0^{(2)}$. For $c_0^{(j)}$, repeatedly using Lemma 5.1, we can calculate

$c_i^{(j)}$, for $i = 1,2,..., p$. We can also find two solutions $c^{(1)}(x)$ and $c^{(2)}(x)$.

We can analyze the complexity of the algorithm. To find each c_i, we need to calculate a convolution of $\sum_{i=0}^{t} c_i d_{t-i}$. The total complexity is $O(p^2)$, i.e., $O(n^2)$.

Now let us consider the case of $s = 3$. Let

$$Q(x,y) = m(x)y^3 + e(x)y^2 + d(x)y + n(x) = (y + C(x))(m(x)y^2 + k(x)y + h(x)) \ . \qquad (4.6)$$

Multiplying (4.6) by $m(x)^2$ and denoting $[m(x)y]$ by z, we have

$$z^4 + E(x)z^2 + D(x)z + N(x) = 0 \ , \qquad (4.6')$$

where $E(x) = e(x)^2 + m(x)d(x)$, $D(x) = e(x)m(x)d(x) + m(x)^2 n(x)$, and $N(x) = e(x)m(x)^2 n(x)$. Thus, it needs only consider the factorization of $Q(x,y) = y^2 + E(x)y^2 + D(x)y + N(x)$. If $(y - c(x))$ is a factor of $Q(x,y)$, we have

$$c(x)^4 + E(x)c(x)^2 + D(x)c(x) + N(x) = 0 \ , \qquad (4.7)$$

i.e.,

$$\sum_{i=0}^{p} c_i^4 x^{4i} + (\sum_{i=0}^{2p} e_i x^i)(\sum_{i=0}^{p} c_i^2 x^{2i}) + (\sum_{i=0}^{3p} d_i x^i)(\sum_{i=0}^{p} c_i x^i) + \sum_{i=0}^{4p} n_i x^i = 0 \ , \qquad (4.7')$$

where $E(x) = \sum_{i=0}^{2p} e_i x^i$, $D(x) = \sum_{i=0}^{3p} d_i x^i$, and $N(x) = \sum_{i=0}^{4p} n_i x^i$. Assume that $d_0 = \cdots = d_{\sigma-1} = 0$, $d_\sigma \neq 0$, and $e_0 = \cdots = e_{\pi-1} = 0$, $e_\pi \neq 0$, for $\sigma, \pi \geq 0$. From (4.7'), we have

$$\begin{cases} (c_{\frac{t}{4}})^4 + \sum\limits_{i=0}^{\frac{t}{2} - \lfloor \frac{\pi}{2} \rfloor} c_i^2 e_{t-2i} + \sum\limits_{i=0}^{t-\sigma} c_i d_{t-i} + n_t = 0 & \text{for } t = 4k \qquad (a') \\ \sum\limits_{i=0}^{\frac{t}{2} - \lfloor \frac{\pi}{2} \rfloor} c_i^2 e_{t-2i} + \sum\limits_{i=0}^{t-\sigma} c_i d_{t-i} + n_t = 0 & \text{for } t = 4k+2 \qquad (b') \\ \sum\limits_{i=0}^{\frac{t-1}{2} - \lfloor \frac{\pi}{2} \rfloor} c_i^2 e_{t-2i} + \sum\limits_{i=0}^{t-\sigma} c_i d_{t-i} + n_t = 0 & \text{for } t \textit{ is odd} \qquad (c') \end{cases} \ , \qquad (4.8)$$

By the same approach, we have:

Lemma 4.2: If $c_0, c_1, ..., c_{\lambda-1}$ are known and $\lambda > \max \{ \frac{\sigma}{3}, \sigma - 2 \times \lfloor \frac{\pi}{2} \rfloor \}$, then c_λ can be calculated by (4.8).

Now we discuss how to calculate $c_0, c_1, ..., c_\kappa$, where κ denotes $\max \{ \frac{\sigma}{3}, \sigma - 2 \times \lfloor \frac{\pi}{2} \rfloor \}$. First we set $t = 0$. Then from Eq. (a') we can calculate c_0. If (1) $\sigma \neq 0$ and $\pi \neq 0$, then Eq. (a') has an unique solution c_0; (2) if $\sigma \neq 0$ and $\pi = 0$, it has at most two solution c_0; (3) if $\sigma = 0$, then it has at most four solutions. Suppose we find $c_0, ..., c_{\mu-1}$, where $1 \leq \mu \leq \kappa$. Now we want to find c_μ. We discuss it in the following. Let $1 \leq \mu \leq \kappa = \max \{ \frac{\sigma}{3}, \sigma - 2 \times \lfloor \frac{\pi}{2} \rfloor \}$. We consider two cases:

Case 1: $\frac{\sigma}{3} \geq \sigma - 2 \times \lfloor \frac{\pi}{2} \rfloor$, i.e., $\lfloor \frac{\pi}{2} \rfloor \geq \frac{\sigma}{3}$.

In this case, since $\mu \le \kappa = \frac{\sigma}{3}$ we have $\mu \le \lfloor \frac{\pi}{2} \rfloor$. Let $t = 4\mu$ and consider Eq. (a'). We have $\frac{t}{2} - \lfloor \frac{\pi}{2} \rfloor \le \mu$ and $t - \sigma \le \mu$, because $\mu \le \lfloor \frac{\pi}{2} \rfloor$ and $\mu \le \frac{\sigma}{3}$. The equality is true only for $\mu = \kappa$. Thus, in Eq. (a'), there is only c_μ^4 and the other c_i are with $i < \mu$. Eq. (a') has an unique solution of c_μ. When $\mu = \kappa$, in Eq. (a'), there are c_μ^4, c_μ^2 and, c_μ, and other c_i with $i < \mu$. Thus, Eq. (a') has at most four solutions of c_μ, i.e., c_κ.

Case 2: $\frac{\sigma}{3} < \sigma - 2\times\lfloor \frac{\pi}{2} \rfloor$, i.e., $\lfloor \frac{\pi}{2} \rfloor < \frac{\sigma}{3}$.

We consider the following cases:

Subcase 2.1: $\mu + \lfloor \frac{\pi}{2} \rfloor$ is odd.

Let $t = 2\mu + 2\times\lfloor \frac{\pi}{2} \rfloor$, using Eq. (b'), we have $\frac{t}{2} - \lfloor \frac{\pi}{2} \rfloor = \mu \ge t - \sigma$, because $\mu \le \sigma - 2\times\lfloor \frac{\pi}{2} \rfloor$. The equality is only true for $\mu = \kappa = \sigma - 2\times\lfloor \frac{\pi}{2} \rfloor$. Thus, when $\mu < \kappa$, Eq. (b') has an unique solution of c_μ; when $\mu = \kappa$, Eq. (b') has two solutions of c_μ.

Subcase 2.2: $\mu + \lfloor \frac{\pi}{2} \rfloor$ is even.

If $\mu > \lfloor \frac{\pi}{2} \rfloor$, then let $t = 2\mu + 2\times\lfloor \frac{\pi}{2} \rfloor$. Using (a'), we have $\frac{t}{2} - \lfloor \frac{\pi}{2} \rfloor = \mu > \frac{\mu + \lfloor \frac{\pi}{2} \rfloor}{2} = \frac{t}{4}$ and $\frac{t}{2} - \lfloor \frac{\pi}{2} \rfloor = \mu \ge 2\mu + 2\times\lfloor \frac{\pi}{2} \rfloor = t - \sigma$, because $\mu \quad \kappa = \sigma - 2\times\lfloor \frac{\pi}{2} \rfloor$. Thus Eq. (a') has an unique solution of c_μ when $\mu < \kappa$ and has two solutions when $\mu = \kappa$.

If $\mu < \lfloor \frac{\pi}{2} \rfloor$, then let $t = 4\mu$. Using Eq. (a'), we have $\frac{t}{4} = \mu > 2\mu - \lfloor \frac{\pi}{2} \rfloor = \frac{t}{2} - \lfloor \frac{\pi}{2} \rfloor$ and $\frac{t}{4} = \mu > 4\mu - \sigma = t - \sigma$, because $\mu < \lfloor \frac{\pi}{2} \rfloor < \frac{\sigma}{3}$. Thus, Eq. (a') has an unique solution of c_μ.

If $\mu = \lfloor \frac{\pi}{2} \rfloor$, let $t = 2\times(\mu + \lfloor \frac{\pi}{2} \rfloor + 1$. Then, t is odd, using (c') we have $\frac{t-1}{2} - \lfloor \frac{\pi}{2} \rfloor = \lfloor \frac{\pi}{2} \rfloor = \mu > 2\times\lfloor \frac{\pi}{2} \rfloor + 2\mu + 1 - \sigma < \mu$, because $\sigma - 2\times\lfloor \frac{\pi}{2} \rfloor > \frac{\sigma}{3} \ge \lfloor \frac{\pi}{2} \rfloor + 1 = \mu + 1$. Thus, Eq. (c') has an unique solution of c_μ.

From the above discussion, when $\mu < \kappa$, the c_μ has an unique solution and when $\mu = \kappa$, the c_μ may have at most four solutions (in case Case 2, c_μ has at most two solutions), provided $c_0, c_1, \ldots, c_{\mu-1}$ are known. Then repeatedly using Lemma 5.2, c_j, for $j = \kappa+1, \ldots, p$ can be uniquely calculated. On the other hand, if $\mu = \kappa \ne 0$, then $\sigma \ne 0$. Thus, (1) if $\sigma \ne 0$ and $\pi \ne 0$, then c_0 has an unique solution. and c_κ has at most four solutions; (2) if $\sigma \ne 0$ and $\pi = 0$, the Case 1 is impossible and in Case 2, c_κ has at most two solutions and c_0 has at most two solutions. Hence, c_0 and c_κ have at most four solutions, and other c_i has an unique solution. Therefore, we have at most four solutions $c(x)$ by the algorithm.

To determine the complexity of the algorithm, for c_0 and c_κ, we need to find roots of equations with degree 2 or an equation with degree 4. When the equation with degree 4, in these equations there are only x, x^2, and x^4, the complexity of finding roots is lower. In finding each c_i we need to calculate at most two convolutions with length p or less. The total complexity is $O(p^2) = O(n^2)$.

V. Conclusions

In this paper, we have derived a generalized Welch-Berlekamp algorithm for the weighted curve fitting problem using the fundamental iterative algorithm in [15]. The complexity is $O(sn^2)$ and the required memory is $O(n)$. When $s = 1$, the refined algorithm reduces to the well-known Welch-Berlekamp algorithm. We also derived a fast factorization for bivariate polynomial with y-degree 2 or 3 over $GF(2^m)$. The complexity is $O(n^2)$. Considering the application of the generalized Welch-Berlekamp algorithm and fast factorization algorithm to the decoding of Reed-Solomon codes over $GF(2^m)$ beyond the true minimum distance, the total complexity is $O(n^2)$. It is equal to the complexity of traditional decoding algorithm.

Acknowledgment

The author is deeply grateful to Prof. R. E. Blahut, Dr. W. S. Feng, and Mr. C. H. Lu for helpful discussion. The author would like to thank Prof. T. Hoholdt for providing his reprint paper [16], which presents the nearly same algorithm based on the fundamental iterative algorithm in [15].

References

[1] V. Guruswami and M. Sudan, "Improved Decoding of Reed-Solomon and Algebraic-Geometric Codes," Preprint, 1998.

[2] L. R. Welch and E. R. Berlekamp, "Error correction of algebraic block codes," *US Patent* Number 4,633,470, issued December 1986.

[3] E. R. Berlekamp, "Bounded Distance +1 Soft-Decision Reed-Solomon Decoding," *IEEE Trans. on Information Theory* Vol. 42, no. 3, pp. 704-720, May 1996.

[4] S. R. Blackburn, "The Welch-Berlekamp and Berlekamp-Massey Algorithm," *Proceedings of 1995 IEEE International Symposium of Information Theory,* Whistler, Canada, pp.409-409, Sept. 1995.

[5] A. C. Antoulas, B. D. Q. Anderson, "On the Scalar Rational Interpolation Problem," *IMA J, Math, control and Inform,* Vol. 3. pp. 61-88, 1986.

[6] W. G. Chambers, R. E. Peile, K. Y. Tsie, N. Zein, "Algorithm for Solving the Welch-Berlekamp Key Equation, with a Simplified Proof." *Elect. Letters* Vol. 29, No.18, pp. 1620-1621, Sept. 1993.

[7] Ding-Jia Xin, "Extension of Welch-Berlekamp Theorem with Applications", *Science in China (Series A)* Vol.38, No.11., Nov. 1995.

[8] M. Sudan, "Decoding Reed-Solomon Codes beyond the Error-Correction Diameter," *Proc. 35th Annual Allerton Conference on Communication, Control and Computing, 1997* pp. 215-224, 1997.

[9] M. Sudan, "Decoding Reed-Solomon Codes beyond the Error-Correction bound," *Journal of Complexity*, 13(1), pp.180-193, March 1997.

[10] W. S. Feng and R. E. Blahut, "Some Results on the Sudan Algorithm," *Proceedings of 1998 IEEE International Symposium of Information Theory,* MIT, USA, pp.57-57, Aug. 1998.

[11] W. S. Feng and R. E. Blahut, "An Analysis of the Sudan Algorithm," Preprint.

[12] R. M. Roth and G. Ruckenstein, "Efficient Decoding of Reed-Solomon Codes Beyond Half the Minimum Distance", *Proceedings of 1998 IEEE International Symposium of Information Theory,*

MIT, USA, pp.57-57, Aug. 1998.

[13] I. Duursma, "Interpolation and Approximation in Decoding," *Proceedings of 1998 IEEE Information Theory Workshop,* Killarney, Ireland, pp.37-37, June 1998.

[14] M. A. Shokrollahi and H. Wasserman, "Decoding Algebraic Geometric Codes," *Proceedings of 1998 IEEE Information Theory Workshop,* Killarney, Ireland, pp.40-41, June 1998.

[15] G. L. Feng and K. K. Tzeng,"A generalization of the Berlekamp-Massey algorithm for multisequence shift-register synthesis with applications to decoding cyclic codes," *IEEE Trans. Infor. Theory,* Vol IT-37, pp. 1274-1287, Sept. 1991.

[16] R. R. Nielsen and T. Hoholdt, "Decoding Reed-Solomon codes beyond half the minimum distance," to appear in the proceeding of the ICCC'98.

From weight enumerators to zeta functions *

Iwan Duursma
Dept. of Mathematics, University of Limoges
123, Ave Albert Thomas, 87100 Limoges, France
duursma@unilim.fr

Abstract

In [1], we introduced the zeta function of a linear code as a tool to describe and study its weight distribution. We present here an explicit transformation from the weight enumerator of a code into its zeta function. And we formulate a conjectural upper bound for the parameters of self-dual linear codes.

Let C be a linear code of length n and minimum distance d over the finite field of q elements. Let A_i be the number of words of weight i in C. The weight enumerator of the code C is defined as

$$A_C(x,y) = x^n + \sum_{i=d}^{n} A_i x^{n-i} y^i$$

To avoid dealing with degenerate cases we will assume throughout that both C and its dual $C^\perp$ have effective length n, or equivalently, that both have minimum distance at least two.

In [1] we defined, for a weight enumerator of length n and minimum distance d over the finite field of q elements its zeta polynomial $P(T)$. It is the unique polynomial of degree at most $n-d$ such that

$$\frac{P(T)}{(1-T)(1-qT)}\,(y(1-T)+xT)^n \quad = \quad \cdots + \frac{A(x,y)-x^n}{q-1}\,T^{n-d} + \cdots$$

In the setting of algebraic curves, that served as a motivation for the definition, the properties of the polynomial $P(T)$ are well understood. It is of degree $2g$, where g is the genus of the curve. And the number of points N on the curve equals $N = q+1+a$, where $-a$ is the sum of the reciprocal roots of $P(T)$. It was proved by A. Weil in 1948 that the reciprocal zeros of the zeta polynomial of an algebraic curve all have absolute value $\sqrt{q}$, with important consequences for the number of rational points on an algebraic curve. For linear codes, if we write $P(T) = a_0(1 + aT + \cdots)$, we find

$$\frac{A_{d+1}}{q-1} = a_0\binom{n}{d+1}(q+1+a) - a_0\binom{n}{d+1}(d+1)$$

It follows that $d+1 \leq q+1+a$ and estimates for the reciprocal zeros of $P(T)$, and in particular for their sum $-a$, yield upper bounds for the minimum distance d.

---

[*] Extended version available from the author.

The given relation between a weight enumerator $A(x, y)$ and its zeta polynomial $P(T)$ can be presented in a different form that is more convenient for the transformation of a weight enumerator into its zeta polynomial.

Theorem For a weight enumerator $A(x, y)$, define

$$a(t) = \frac{1}{q-1} \sum_{i=d}^{n} A_i / \binom{n}{i} t^{i-d}$$

We have

$$\frac{P(T)}{(1-T)(1-qT)}(1-T)^{d+1} \equiv a(\frac{T}{1-T}) \pmod{T^{n-d+1}}$$

The function $Z(T) = P(T)/((1-T)(1-qT))$ is called the zeta function of the linear code. Using properties of the zeta function of a curve, Drinfeld and Vladuts [2] obtained the following optimal asymptotic upper bound for the number of rational points N on a curve relative to the genus g of the curve.

$$\lim_{g \to \infty} \sup N/g \leq \sqrt{q} - 1.$$

For self-dual codes, the zeta polynomial $P(T)$ is of degree $n + 2 - 2d$. And in the analogy between curves and self-dual codes, the parameters N and g correspond to $d + 1$ and $n/2 + 1 - d$. Under the assumption that the zeta function of a code has properties similar to the zeta function of a curve, the Drinfeld-Vladuts bound translates into an upper bound on the relative distance.

Conjecture For an infinite family of self-dual codes of increasing length,

$$\lim_{n \to \infty} \sup d/n \leq 1/2 - 1/(2\sqrt{q})$$

For small values of q, we compare with the known upper bounds [3], [4], [5]. Note that for $q \to \infty$ the upper bound is correct.

$$\begin{array}{lll} (q=2) & 1/2 - 1/(2\sqrt{2}) = 0.146\ldots & (0.165630 < 1/6) \\ (q=3) & 1/2 - 1/(2\sqrt{3}) = 0.213\ldots & (0.25 = 1/4) \\ (q=4) & 1/2 - 1/(2\sqrt{4}) = 0.25 & (0.333\ldots = 1/3) \end{array}$$

References

[1] I. M. Duursma, Weight distributions of geometric Goppa codes, Trans. Amer. Math. Soc. 351 (1999), pp. 3609-3639.

[2] S. G. Vladuts and V. G. Drinfeld, Number of points of an algebraic curve, Func. Anal. 17 (1983), pp. 53-54.

[3] C. L. Mallows, A. M. Odlyzko, and N. J. A. Sloane, Upper bounds for modular forms, lattices and codes, J. Algebra 36 (1975), pp. 68-76.

[4] E. Rains, Shadow bounds for self-dual codes, IEEE Trans. Info. Theory, to appear.

[5] I. Krasikov and S. Litsyn, An improved upper bound on the minimum distance of doubly-even self-dual codes, IEEE Trans. Info. Theory, to appear.

Alternative Approaches to the Computation of Error Values for Hermitian Codes*

Michael E. O'Sullivan
Mathematics Dept.
National University of Ireland
Cork, Ireland
michael.osullivan@ucc.ie

Abstract

Kötter's algorithm for decoding one-point codes produces a set of error locator polynomials. The advantages of the algorithm are that the logical control is extremely simple and that it uses several Berlekamp-Massey machines working in parallel. In this article we show that minor modifications to the algorithm will produce error evaluator polynomials as well. We also generalize an error evaluation formula for Reed-Solomon codes due to Kötter which involves the derivatives of the locator polynomials and the updating polynomials.

1 Introduction

Kötter's algorithm for decoding one-point codes uses several Berlekamp-Massey type algorithms running in parallel and is the most efficient algorithm for hardware implementation currently available. We show here that with only minor modifications Kötter's algorithm for decoding Hermitian codes can be used to compute error evaluator polynomials in addition to the locator polynomials. The coefficients of the evaluator polynomials are actually already computed in the current formulation of the algorithm, but the information is not used.

Unfortunately, the extra memory required to store the evaluator polynomials makes their computation less practical then simply computing one evaluator after the iterative algorithm is complete. There is an alternative method for error evaluation that uses the update polynomials (which failed at a previous iteration) which may prove to be computationally more efficient than current methods.

---

*This research is supported in part by NSF Grant No. CCR-98-05080

This article will survey results in three main subjects which are treated more extensively in the referenced articles, (in particular [2, 4, 7]): the key equation, Kötter's algorithm, and the use of update polynomials in error evaluation.

2 Hermitian Codes

Let $\mathbb{F}_q$ denote the finite field with q elements. We will consider codes constructed from the Hermitian curve $\mathcal{H}$ over $\mathbb{F}_{q^2}$ defined in the xy-plane by

$$y^q + y = x^{q+1}$$

We now review some important facts about Hermitian curves [8, 9]. There is a unique point Q of $\mathcal{H}$ on the line at infinity. The ring

$$R = \frac{\mathbb{F}_{q^2}[x, y]}{< y^q + y - x^{q+1} >}$$

is the ring of functions on $\mathcal{H}$ which have poles only at Q. Let$K$ be the function field of $\mathcal{H}$. For $f \in K$, let $\nu_Q(f)$ be the valuation of f at Q. We will also use the pole order function $o(f) = -\nu_Q(f)$. Let Λ be the set of pole orders of the elements of R—the nongaps. Then

$$\Lambda = \{aq + b(q+1) : a, b \geq 0\}$$

The Hermitian curve has $n = q^3$ rational points, $P_k = (x_k, y_k)$. For $m \in \mathbb{Z}$ we consider the codes

$$C_m = \{v \in \mathbb{F}_{q^2}^n : \sum v_k f(P_k) = 0 \;\text{ for all } f \in L(mQ)\}$$

where $L(mQ) = \{f \in R : o(f) \leq m\}$. If $e \in \mathbb{F}_{q^2}^n$ is an error vector, the syndrome map defined by e is

$$S_e : R \longrightarrow \mathbb{F}_{q^2}$$
$$f \longmapsto \sum e_k f(P_k)$$

When encoding with C_m, we receive a vector $v + e$ with $v \in C_m$, so we only know the action of the syndrome map on the space $L(mQ)$. The decoding question becomes how to process the knowledge of S_e on $L(mQ)$ to obtain the vector e. For the purpose of exposition we will just focus on the question of computing e from S_e (on all of R). In practice the Feng-Rao majority vote is used to compute $S_e|_{L((m+1)Q)}$ from $S_e|_{L(mQ)}$. Thus if e is correctable, S_e is computable.

3 Two Sets of Bases for the Function Field

We now identify two different ways to write elements of the function field K which are dual to each other. The first basis extends the basis for R used in Kötter's algorithm to all of K.

For any $u \in \mathbb{Z}$ there exist unique integers a, b such that $u = aq + b(q+1)$ and $0 \leq b < q$. We define

$$z_u = x^a y^b$$

Then $o(z_u) = u$ and

$$u \in \Lambda \Longleftrightarrow a \geq 0 \Longleftrightarrow z_u \in R$$

Furthermore, any element $f \in K$ has a unique expression as a series

$$f = \sum_{u \geq \nu_Q(f)} a_u z_{-u} \tag{1}$$

For the other basis we again write $u = aq + b(q+1)$ with $0 \leq b < q$. We define

$$\zeta_u = \begin{cases} x^{-a-1}(y^{q-1}+1) & \text{if } b = 0 \\ x^{-a-1}y^{q-1-b} & \text{if } b \neq 0 \end{cases} \tag{2}$$

In this case

$$\begin{aligned} \nu_Q(\zeta_u) &= (-a-1)(-q) + (q-1-b)(-q-1) \\ &= aq + b(q+1) - q^2 + q + 1 \\ &= u - (2g-1) \end{aligned}$$

where g is the genus of $\mathcal{H}$. Furthermore,

$$u \in \Lambda^c \Longleftrightarrow a < 0 \Longleftrightarrow \zeta_u \in R$$

As with the z_u, any $f \in K$ has a unique expansion in terms of the ζ_u.

Let dx be the differential of x. The divisor (dx) is $(2g-2)Q$ so $\nu_Q(\zeta_u dx) = u - 1$. Since $\nu_Q(z_u) = -u$, we have that $\nu_Q(z_u \zeta_u dx) = -1$ and therefore its residue at Q is nonzero. The following theorem show that the two basis $\{z_u\}$ and $\{\zeta_u\}$ are dual with respect to this operation of multiplying by dx and taking the residue at Q.

Theorem 3.1 **([4, Ex. 6.5]).** *For any $u, v \in \mathbb{Z}$,*

$$\operatorname{res}_Q z_u \zeta_v dx = \begin{cases} -1 & \text{if } v = u \\ 0 & \text{otherwise} \end{cases}$$

4 The Syndrome Series and the Key Equation

For u a nongap, let $s_u = S_e(z_u)$. Then if $f = \sum_{u \in \Lambda} a_u z_u$, by linearity we have $S_e(f) = \sum_{u \in \Lambda} a_u s_u$.

Define the syndrome series as

$$h_e = \sum_{u\in\Lambda} s_u\zeta_u$$

It is clear from Theorem 3.1, that $s_u = \mathrm{res}_Q(z_u h_e dx)$ so the action of $S_e(\star)$ and $\mathrm{res}_Q((\star)dx)$ on R are the same. Having the syndrome expressed as a series now lets us identify the product of the syndrome with any $f \in R$. The coefficients of the product are related to syndromes of multiples of f as shown in the following proposition.

Definition 4.1. Let $f \in K$. Re-indexing in (1), we may write $f = \sum_{u\le o(f)} a_u z_u$. We define the R-part of f to be $\sum_{u\in\Lambda} a_u z_u$.

Proposition 4.2. *For $f \in R$, let*

$$fh_e = \sum_{v\in\mathbb{Z}} t_v\zeta_v \tag{3}$$

Then $t_v = S_e(\tilde{f})$ where $\tilde{f}$ is the R-part of fz_v.

PROOF: Write $fz_v = \sum_{u\in\mathbb{Z}} b_u z_u$. Then $\tilde{f} = \sum_{u\in\Lambda} b_u z_u$ and $fz_v = \tilde{f} + \sum_{u\in\Lambda^c} b_u z_u$. From Theorem 3.1, $t_v = -\mathrm{res}_Q\, z_v(fh_e)dx$. But,

$$\mathrm{res}_Q(fz_v)h_e dx = \mathrm{res}_Q\, \tilde{f}h_e dx + \mathrm{res}_Q \sum_{u\in\Lambda^c} b_u z_u h_e dx$$

Since the non-zero terms in h_e are indexed by nongaps, the second term is zero. Thus,

$$\begin{aligned} t_v &= -\mathrm{res}_Q\, \tilde{f}h_e dx \\ &= S_e(\tilde{f}) \end{aligned}$$

□

The following formula provides some additional justification for the use of the series h_e to represent the syndrome. Let h_k be the syndrome series for the kth basis vector which is 1 in the kth place and 0 elsewhere. Then $h_e = \sum e_k h_k$.

$$\begin{aligned} h_k &= \sum_{u\in\Lambda} z_u(P_k)\zeta_u \\ &= \sum_{\substack{0\le a \\ 0\le b<q}} x_k^a y_k^b x^{-a-1}(y^{q-b-1} + \delta_b) \end{aligned}$$

where δ_b is 1 if $b = 0$ and 0 otherwise,

$$\begin{aligned} &= \frac{1}{x}\sum_{0\le a}(\frac{x_k}{x})^a + \frac{y^{q-1}}{x}\sum_{0\le a}(\frac{x_k}{x})^a \sum_{0\le b<q} \frac{y_k}{y})^b \\ &= \frac{1}{x-x_k} + \frac{1}{x-x_k}\cdot\frac{y^q-y_k^q}{y-y_k} \\ &= \frac{1}{x-x_k}\cdot\frac{y^q-y-y_k^q-y_k}{y-y_k} \end{aligned}$$

Thus,

$$h_e = \sum_k e_k \frac{1}{x - x_k} \cdot \frac{y^q - y - y_k^q - y_k}{y - y_k}$$

This formula is essentially what is used for error evaluation in [1, §3]. It is easily verified that h_k has a pole of order 1 at P_k and at Q and no other poles, and that $\operatorname{res}_P h_k dx = -res_Q h_k dx = 1$. Now it is clear that $f \in R$ is an error locator if and only if fh_e has no poles except at Q. This leads to the key equation.

Definition 4.3. We say that $f, \varphi \in R$ satisfy the *key equation* if

$$fh_e = \varphi$$

Proposition 4.4. *The following are equivalent:*

1. *f satisfies the key equation.*
2. *$f(P_k) = 0$ at all error positions ($k : e_k \neq 0$).*
3. *$S_e(fz_u) = 0$ for all $u \in \Lambda$.*

The extension of Kötter's algorithm computes successive approximations to the key equation.

Definition 4.5. For $f \in R$ we define $\operatorname{span} f$ to be the smallest nongap c such that $S_e(z_c f) \neq 0$ (or $\operatorname{span} f = \infty$ if no such c exists). We say that $f, \varphi \in R$ satisfy the *nth key equation* if φ is a linear combination of the ζ_v with $v \in \Lambda^c$ and $v \leq n$, and

$$\nu_Q(fh_e - \varphi) \geq n - 2g + 2$$

The *discrepancy of f, φ at $n + 1$* is the unique $\alpha \in k$ such that

$$\nu_Q(fh_e - \varphi - \alpha\zeta_{n+1}) > n - 2g + 2 \tag{4}$$

Note that the definition allows for $f = 0$. For example, if $v \in \Lambda^c$ then $0, \zeta_v$ satisfy the $(v-1)$th key equation and the discrepancy is -1 at v.

5 Kötter's Algorithm with Computation of Error Evaluator Polynomials

We now present a matrix formulation for solving the key equation using Kötter's algorithm. Proofs of the results in this section are given for all one-point codes in [7].

Certain subtleties of implementation are not included in the algorithm. They are amply discussed in the references, so we limit ourselves to a few brief comments. Let the encoding code be $C_{m^\star}$. The algorithm has iterations for all integers $m = 0$ to M where M is generally larger than $m^\star$. In practice majority voting is included in step 1 when $m > m^\star$ [2]. If majority voting is unsuccessful, or if the size of the δ-set [6] gets to large, decoding is aborted. The last iteration, M, is determined by the weight and the δ-set of the error vectors to be corrected[5]. For high-rate codes, error vectors of weight much larger than the minimum distance can be corrected provided the distribution of errors is "random."

Data Structure For $i = 0, \ldots, q-1$,

$$B_i^{(m)} = \begin{bmatrix} f_i & \varphi_i \\ g_i & \psi_i \end{bmatrix}^{(m)}$$

Initialization

$$B_0^{(-1)} = \begin{bmatrix} 1 & 0 \\ 0 & -(y^{q-1}+1) \end{bmatrix}$$

and for $i \neq 0$,

$$B_i^{(-1)} = \begin{bmatrix} y^i & 0 \\ 0 & -y^{q-1-i} \end{bmatrix}$$

Algorithm For $m = 0$ to M.

1. Compute the discrepancies,
   Let $s_i = \deg f_i$, let
   $$\alpha_i = S_e(\tilde{f}_i)$$
   where $\tilde{f}_i$ is the R-part of $f z_{m-s_i}$.
2. For each $i = 0, \ldots, q-1$, let j be the least positive residue of $m - i$ modulo q
   Set $r_i = 1 + (m - s_i - s_j)/q$
   If $r_i \leq 0$ or $\alpha_i = 0$, set
   $$A_i = \begin{bmatrix} 1 & -\alpha_i x^{-r_i} \\ 0 & 1 \end{bmatrix} \tag{5}$$
   If $r_i > 0$, set
   $$A_i = \begin{bmatrix} x^{r_i} & -\alpha_i \\ \alpha_i^{-1} & 0 \end{bmatrix} \tag{6}$$
   Then
   $$\begin{bmatrix} f_i^{(m)} & \varphi_i^{(m)} \\ g_j^{(m)} & \psi_j^{(m)} \end{bmatrix} = A_i \begin{bmatrix} f_i^{(m-1)} & \varphi_i^{(m-1)} \\ g_j^{(m-1)} & \psi_j^{(m-1)} \end{bmatrix}$$

We now summarize the properties of the data computed.

Theorem 5.1. *Upon completion of the mth iteration of the algorithm:*

1. $o(f_i^{(m)}) \cong i \mod q$, *and* $f_i^{(m)}$ *has leading coefficient 1;*
2. $f_i^{(m)}$ *and* $\varphi_i^{(m)}$ *satisfy the* $m - o(f_i^{(m)})$ *key equation;*
3. $o(g_i^{(m)}) \leq m - o(f_i^{(m)}) + q$;
4. $g_i^{(m)}$ *and* $\psi_i^{(m)}$ *satisfy the* $o(f_i^{(m)}) - q - 1$ *key equation with discrepancy* 1 *at* $o(f_i^{(m)}) - q$; *and*
5. $o(f_i^{(m)}) = i(q+1)$ *if and only if* $g_i^{(m)} = 0$, *in which case* $\psi_i^{(m)} = -(y^{q+1-i})$ *if* $i \neq 0$ *or* $\psi_i^{(m)} = -(y^{q+1}+1)$ *if* $i = 0$.

Corollary 5.2. *For* M *large enough,* $f_i^{(M)}$, $\varphi_i^{(M)}$ *satisfy the key equation and* $f_i^{(M)}$ *is the function in* R *of smallest pole order congruent to* i *modulo* q *to do so. The* $f_i^{(M)}$ *generate* I_e*, the ideal of functions vanishing at the error positions.*

6 Error Evaluation Using Error Evaluators

We now consider two methods for error evaluation. We will drop the superscripts on the functions computed in the algorithm, so $f_i = f_i^{(M)}$.

The $\{f_i\}$ form a generating set for I_e, so for any P_k there is some f_i which has a zero of multiplicity 1 at P_k. Therefore we can use the following proposition to evaluate errors.

Proposition 6.1. *If* f *is a locator with multiplicity one at an error position* P_k *then*

$$e_k = \frac{\varphi(P_k)}{f'(P_k)} \tag{7}$$

where $f' = df/dx$.

PROOF: We will show that for any P_k at which f vanishes,

$$\varphi(P_k) = e_k f'(P_k) \tag{8}$$

If f has a zero of multiplicity one we may divide by $f'(P_k)$ and get (7). Recall that $h_e = \sum_l e_l h_l$ with h_l having poles only at P_l and Q and those of order 1. Since f vanishes at P_k, $fh_l(P_k) = 0$ for $l \neq k$. Thus $\varphi(P_k) = e_k(fh_k)(P_k)$. Now $x - x_k$ is a uniformizing parameter at P_k so we may expand f_i and h_k with respect to $x - x_k$. We get

$$h_k = (x - x_k)^{-1} + \cdots$$
$$f_i = a_1(x - x_k) + a_2(x - x_k)^2 + \cdots$$

Now $(fh_k)(P_k) = a_1$ which equals $df/dx(P_k)$ as was to be proved. $\square$

The downside of using this method is that one may have to check several of the f_i before finding one which has a zero of multiplicity 1 at a given P_k.

Another method is used in the current implementation of Kötter's algorithm. Let f be the error locator of lowest pole order. The zeros of f are computed (this may introduce some extraneous locations). The error values are found using (7) by applying a version of L'Hopitals rule with Hasse derivatives if f has a higher order zero at P_k. If P_k is an extraneous position ($e_k = 0$), $\varphi(P_k)$ will be zero with the same multiplicity as f.

In the current implementation, the error evaluators are not computed iteratively, which saves an enormous amount of memory (the size of $2q$ polynomials of length roughly $t+2g-1$). The product fh_e is computed by convolving the syndrome array and f. This, unfortunately requires some delay, since there are approximately $t + 2g$ coefficients.

7 Evaluation Using Update Polynomials

A third alternative for computing error values uses the theory developed here and an observation of Kötter's [3, Th. 2]. In the Berlekamp-Massey algorithm the matrices involved in the update of the locator-evaluator candidates all have determinant 1. Thus the determinant of the data set

$$\begin{bmatrix} f^{(m)} & \varphi^{(m)} \\ g^{(m)} & \psi^{(m)} \end{bmatrix}$$

is invariant (equal to -1). After the final iteration (dropping superscripts) $f\psi - g\varphi = -1$. Evaluating at an error position we get $g(P_k)\varphi(P_k) = 1$. Since $\varphi(P_k) = e_k f(P_k)$,

$$e_k^{-1} = f'(P_k)g(P_k)$$

and there is no need to compute error evaluators.

We now show how this is generalized to Hermitian codes. Proofs of these results are given in [7].

Theorem 7.1. *At each mth iteration of the algorithm,*

$$\sum_{i=0}^{q-1} \det B_i^{(m)} = -1$$

The proof requires the following proposition.

Proposition 7.2. *In the mth iteration of the algorithm, $\alpha_i = \alpha_j$ whenever $i + j \cong m$* mod q.

Furthermore, for any m,

$$\operatorname{res}_Q \det \begin{bmatrix} f_i^{(m)} & \varphi_i^{(m)} \\ g_j^{(m)} & \psi_j^{(m)} \end{bmatrix} x^a = \begin{cases} 1 & \textit{if } i = j \textit{ and } a = -1 \\ 0 & \textit{otherwise} \end{cases} \tag{9}$$

$$\operatorname{res}_Q \det \begin{bmatrix} f_i^{(m)} & \varphi_i^{(m)} \\ f_j^{(m)} & \varphi_j^{(m)} \end{bmatrix} x^a = 0 \tag{10}$$

$$\operatorname{res}_Q \det \begin{bmatrix} g_i^{(m)} & \psi_i^{(m)} \\ g_j^{(m)} & \psi_j^{(m)} \end{bmatrix} x^a = 0 \tag{11}$$

The theorem gives a formula for computing error values. Dropping the superscript M on $\det B_i^{(M)}$, we have

$$\sum_{i=0}^{q-1} f_i \psi_i - g_i \varphi_i = -1$$

Evaluating at an error position P_k, the left term vanishes so by (7)

$$e_k^{-1} = \sum_{i=1}^{q-1} g_i(P_k) f'(P_k) \tag{12}$$

8 Comparison of Methods

We now compare the Kötter's method with using (12) for computing error values. First, we remark that Proposition 7.2 already has some impact on implementation. It shows that the ith and jth discrepancies are equal if $i + j \cong m \mod q$. Thus the number of discrepancies to be computed is roughly halved ($(q+1)/2$ if q odd, $1 + q/2$ if q, m even, $q/2$ if q even m odd) which leads to a reduction of circuitry.

Using the new method requires finding the exact set of error locations. Since the lowest pole order locator has at most $t + g$ zeros and the others need only be evaluated at these points, this involves at most $n + (t + g)(q - 1)$ polynomial evaluations. If we take $t \approx g \approx q^2/2$ and $n = q^3$, this gives roughly $2n$ polynomial evaluations as compared to n evaluations required to find the zeros of just one locator.

On the other hand, we avoid the computation of the error evaluator φ which requires $t + 2g \approx 3q^2/2$ discrepancy computations. These are in some ways more costly than polynomial evaluations since Horner's method may be used for the latter. The computation of error values is more regular with the new method since it avoids evaluating higher order derivatives when f has a higher order zero at P_k.

To summarize, there are advantages to each method and choosing the optimum method will require a closer analysis of hardware issues and may also depend on the number of errors to be corrected.

Acknowledgement

The author had many productive conversations with Ralf Kötter which led to the current formulation of the algorithm for solving the key equation algorithm.

References

[1] J. P. Hansen, H. E. Jensen, R Kötter, "Determination of error values for algebraic geometry codes and the Forney formula," *IEEE Trans. Inform. Theory*, vol. IT-44, pp. 1881-1886, Sept. 1998.

[2] R. Kötter, "A fast parallel implementation of a Berlekamp-Massey algorithm for algebraic-geometric codes," *IEEE Trans. Inform. Theory*, vol. IT-44, pp. 1353-1368, July, 1998.

[3] R. Kötter, "On the determination of error values for codes from a class of maximal curves," in *Proc. 35th Allerton Conf. on Communication, Control and Computing*, Univ. Illinois, 1997.

[4] M. E. O'Sullivan, "The key equation for one-point codes and efficient error evaluation," submitted to *Finite Fields Appl.*

[5] M. E. O'Sullivan, "Decoding of Hermitian codes beyond the minimum distance bound," submitted to *IEEE Trans. Inform. Theory.*

[6] M. E. O'Sullivan, "Decoding of Hermitian codes: the key equation and efficient error evaluation," submitted to *Trans. Inform. Theory.*

[7] M. E. O'Sullivan, "Solving the key equation for one-point codes with Kötter's algorithm," to be submitted to *Trans. Inform. Theory.*

[8] H. Stichtenoth, "A note on Hermitian curves over GF(q^2)," *IEEE Trans. Inform. Theory*, vol. IT-37, pp. 1345-1348, Sept. 1991.

[9] H. J. Tiersma, "Remarks on Codes from Hermitian curves," *Trans. Inform. Theory*, vol. IT-33, pp. 605-609, July, 1987.

Buffer overflow asymptotics in HOL service systems with heterogeneous long-tailed inputs*

Constantinos Kotopoulos
Department of Mathematics
University of Essex
Colchester CO4 3SQ, U.K.

Nikolai Likhanov
Inst. Prob. Info. Trans.
Russian Academy of Sciences
Moscow, Russia

Ravi R. Mazumdar
School of Elec. and Comp. Engr.
Purdue University
W. Lafayette, IN 47907, USA

Abstract

In this paper we obtain the exact buffer overflow asymptotics for buffers fed with heterogeneous long-tailed sources which are served according to a HOL priority scheme. The model we consider is of long-tailed sessions of different rates r_i which arrive to a buffer according to a Poisson process and transmit for periods which have long-tailed distributions with tail decay rate $\sim x^{-(1+\beta_i)}$. These results suggest natural HOL schedules based on source characteristics in such a way that the priority of a given buffer is chosen to minimize the tail probability. These results extend the previous results of the authors on the buffer asymptotics with heterogeneous long-tailed inputs. Moreover, as in the previous work the hypotheses are much weaker than those which have been assumed to derive asymptotics of long-tailed queues with homogeneous inputs by other authors.

1 Introduction

Cell loss asymptotics in buffers fed with long-tailed sources has been of much interest ever since the empirical work of Willinger et al [7] on aggregated LAN traffic and Paxon [12] on internet traffic. By now it is an accepted fact that in order to adequately model internet traffic we haveto depart from conventional light-tailed models such as Markov models.

Cell loss asymptotics have been studied through a surrogate measure corresponding to the probability of exceeding a given level in an infinite buffer queue. This is termed the overflow probability. There have been basically two approaches: the many sources asymptotic which is valid when there are a large number of sources and the large buffer asymptotic which is necessary when buffers are large or there are afew sources. It was shown by Likhanov and Mazumdar [8] that the many sources

---

*This research was supported in part by a grant from Francetelecom CNET and performed while the third author was at the University of Essex

asymptotic remains unchanged in form from the case of light-tailed case provided the instantaneous source rates are finite which is almost always satisfied in practice. The large buffer asymptotics ae however different. In an early paper Duffield [3] obtained upper and lower bounds for the overflow probability when there are longtailed inputs enterring a buffer of infinite capacity. More recently the case of subexponential asymptotics havebeen studied by Jelenkovic and Lazar [5] where they show that the precise asymptotics can be obtained if a certain conjecture holds. The complete proof was recently given by Resnick and Samoridnitsky [14]. A crucial assumption however is that the server capacity is strictly lessthan the rate (taken to be 1) ofa source i.e. one source when transmitting is sufficient instantaneous work to exceed server capacity. A different model based on the $M/G/\infty$ source input model with long-tailed services has been studied by Parulekar and Makowski [11] and by Liu *et al* [10] where they area ble to obtain exact asymptotics only in special cases under the assumption similar to [5]. An excellent state-of-the-art account can befound in Boxma [2].

Recently, the single buffer case with heterogeneous $M/G/\infty$ inputs has been studied by Likhanov and Mazumdar [9] in which the exact buffer overflow as well as cell loss asymptotics were obtained. Moreover the restrictive assumptions of the previous works mentioned above were shown to be unnecessary. When specialized to the homogeneous case and the assumptions of the previous works the result coincides with those reported in the literature. An important outcome was to show that the heterogenous case depends not only on the tail decay rates but also on the source peak rates and it is the ratio which determines the loss characteristics.

The extension of the single buffer case to more complicated service disciplines has only just begun to be addressed. In a recent paper, Borst, Boxma and Jelenkovic [1] studied Generalized Processor Sharing (GPS) queues with long-tailed inputs. They showed that as far as the asymptotics are concerned asort of decoupling takes place, the effects of other long-tailed sources are only felt through their mean rateswhich leads to a reduction in the available capacity by that amount. They could only show their results under the assumption that each queue is individually stable, i.e. the average arriving work is less than the capacity times the weight (GPS) of the queue, and that the peak rates of the sources are identical and are greater than the server capacity.

In this paper we consider the case of buffers which are served according to a HOL priority discipline by which a lower order priority queue is served only when there is no work in the higher priority buffers. These disciplines are important due to the various switch architectures which will be offering the so-called DIFF-SERV services [4]. There are no known results for this case. In the case of two buffers the priority scheme can be seen as the limit of the GPS scheme by letting one of the weights go to 0. However the results of [1] do not apply as the lower priority queue violates the individually stable assumption. The model we consider is a generalization of the $M/G/\infty$ model by which each queue is accessed by different sources with differing peak rates and long-tailed characteristics. Under the assumption that the queueing system is stable we derive the exact overflow probability asymptotics. This yields much insight on how HOL schedules could be designed to obtain optimalcell loss characteristics. Due to space limitations details of proofs are omitted. They can be found in [6].

The organization of this paper is as follows: In section 2 we formulate the model and introduce some notation. In section 3 we recall some important results in the single buffer case which are used crucially in the proof of the more general result. In section 5 we discuss some implications of how the result can be used to define optimal priority orderings.

2 Model and preliminaries

We consider a discrete-time model consisting of N queues served according to the *Head Of the Line* (*H.O.L.*) priority scheme. Particularly, we assume that queue i is fed by type i sessions, $i = 1, 2, \dots, N$ which arrive independently according to a Poisson proces. Once a session of type i arrives it transmits cells at the rate r_i for a random length of time τ_i which has long-tailed distribution G_i given for $z \to \infty$ by $P[\tau_i > z] \sim \alpha_i z^{-1-\beta_i}$ where α_i is a constant and $\beta_i > 0$. From now on we will use the term long-tail to refer to this type of sources. It is assumed that each queue possesses an infinite buffer where the backlogged work can be stored. Under the *H.O.L.* priority discipline a queue is drained only if the server is not being used by queues with higher priority. In our model queue i has priority over the $i+1$ queue. The sessions as well as their lengths in every queue are assumed to be mutually independent.

More precisely, we assume that at any moment t sessions arrive to the i^{th} queue according to a Poisson r.v. with parameter λ_i. Let $\theta_{t,i}$ denote the number of sessions which arrive at the queue i at time t. The r.v.'s $\{\theta_{t,i}\}_t$ are assumed to be i.i.d. and mutually independent among all queues. Then,

$$\mathrm{P}[\theta_{t,i} = n] = \frac{\lambda_i}{n!} e^{-\lambda_i} \, . \tag{2.1}$$

Henceforth the notation $A(z) \sim B(z)$ for $z \to \infty$ will stand for $\lim_{z\to\infty} A(z)/B(z) = 1$.

Let $\tau_{t,i,j}$ denote the length of a typical session arriving at queue i at the moment t and assume that they are i.i.d. Let us moreover denote $Y_{t,i}$ the aggregate number of cells emitted by all active sessions in the i^{th} buffer at the moment t. Then if $\mathbf{1}_{\{A\}}$ denotes the indicator function of the event A we have,

$$Y_{t,i} = \sum_{n=t}^{-\infty} \sum_{j=1}^{\theta_{n,i,j}} r_i \mathbf{1}_{\{\tau_{t,i,j} > t-n\}} \, , \tag{2.2}$$

while the total number of cells entered queue i in $(-t_0, 0)$ will be,

$$X_i(-t_0, 0) = \sum_{t=-t_0}^{-1} Y_{t,i} = \sum_{t=-t_0}^{-1} \sum_{n=-t}^{-\infty} \sum_{j=1}^{\theta_{n,i,j}} r_i \mathbf{1}_{\{\tau_{t,i,j} > t-n\}} \, . \tag{2.3}$$

What is more, for every $\epsilon > 0$ we define the processes $Y^L_{t,i}(\epsilon)$ and $Y^H_{t,i}(\epsilon)$ as follows. The process $Y^L_{t,i}(\epsilon)$ corresponds to the number of cells emitted by sessions which have lengths at most ϵz and we will refer to these as the *light-i* sessions. This is given by,

$$Y^L_{t,i}(\epsilon) = \sum_{n=t}^{t-\epsilon z} \sum_{j=1}^{\theta_n} r_i \mathbf{1}_{\{\epsilon z \ge \tau_{n,j} > t-n\}} \, . \tag{2.4}$$

The process $Y^H_{t,i}(\epsilon)$ corresponds to the number of cells emitted by sessions whose length is greater than ϵz and we will refer to these as the *heavy-i* sessions. This is given by,

$$Y^H_{t,i}(\epsilon) = \sum_{n=t}^{-\infty}\sum_{j=1}^{\theta_n} r_l 1_{\{\tau_{n,j}>\epsilon z\ ,\ \tau_{n,j}>t-n\}}\ . \tag{2.5}$$

then it can be seen that the processes $Y^H_{t,i}(\epsilon)$ and $Y^H_{t,i}(\epsilon)$ are mutually independent and $Y_{t,i} = Y^L_{t,i}(\epsilon) + Y^H_{t,i}(\epsilon)$. Throughout this paper quantities defined with a superscript L or H will denote quantities w.r.t. to the processes defined in 2.4 and 2.5.

We will use the term *heavy-*$(i_1, i_2, \ldots, i_n)$ (resp. *light-*$(i_1, i_2, \ldots, i_n)$) with $n = 1, 2, \ldots N$ for the process $\sum_{k=1}^n Y^H_{t,i_k}$ (resp. $\sum_{k=1}^n Y^L_{t,i_k}$).

Let us further denote the mean number of cells entering buffer i by ρ_i and the total number of cells entering the system by $\rho = \sum_{i=1}^N \rho_i$ i. e.

$$\rho_i = \mathrm{E}[Y_{t,i}] = \lambda_i r_i \mathrm{E}[\tau_{t,i,1}] \tag{2.6}$$

We assume that the system is stable , i.e.; $\rho < C$. Under this condition it is well known from Loynes' theorem that there exists a stationary version of the workload processes.

Let $W_{t,i}$ denote the stationary version of the workload process in the i^{th} queue (i^{th} buffer content at time t). Then by Lindley's recursion,

$$W_{t,i} = \max\{W_{t-1,i} + Y_{t,i} - S_{t,i}, 0\} \tag{2.7}$$

where $S_{t,i}$ stands for the service queue i receives at the time moment t which because of *H.O.L.* service discipline clearly depends on the workload of the other queues. Equivalently, the stationary workload of the i^{th} queue at time 0 can be written as,

$$W_{0,i} = \sup_{t>0}\{X_i(-t,0) - \sum_{s=1}^{t} S_{s,i}\}\ . \tag{2.8}$$

Let us now define the following quantities:

$$J_{i_1,\ldots,i_n} := (j_{i_1}, j_{i_2}, \ldots, j_{i_n}) \tag{2.9}$$

which is a vector of heavy sessions from a collection of $n = 1, 2, \ldots, N$ queues out of the N ones comprising the global system. Further we denote

$$K_{J_{i_1,\ldots,i_n}} := \sum_{u=1}^{n} \beta_i j_i \qquad \text{and} \qquad R_{J_{i_1,\ldots,i_n}} := \sum_{u=1}^{n} r_i j_i\ , \tag{2.10}$$

where the $K_{J_{i_1,\ldots,i_n}}$ corresponds to the decay exponent for the specific collection of heavy-type sessions $J_{i_1,\ldots,i_n}$, while $R_{J_{i_1,\ldots,i_n}}$ is the cell transmission rate associated with the latter collection.

Define:

$$J^0_{i_1,\ldots,i_n} := \min_{J_{i_1,\ldots,i_n}} \{K_{J_{i_1,\ldots,i_n}}\ :\ R_{J_{i_1,\ldots,i_n}} - (C - \sum_{u=1}^{n} \rho_{i_u}) > 0\}$$

where we should note that $J^0_{i_1,\ldots,i_n} \neq (J^0_{i_1}, J^0_{i_2}, \cdots, J^0_{i_n})$ in general, for any $n = 2, 3, \ldots, N$.

Finally let us define the following event,

$A_{J_{i_1,\ldots,i_n}}(0) = \{$There are $J_{i_1,\ldots,i_n}$ heavy-$(i_1, \ldots, i_n)$ active sessions at the instant $0\}$.

3 Preliminary results

Having introduced all necessary notation we proceed by stating some results from a recent paper by Likhanov and Mazumdar ([9]). We should point out that some of the results are not stated here in their full generality but are simplified to meet the requirements of our context. First we define a few quantities in this context.

M denotes the number of types of sources

J denotes a collection of $(j_1, j_2, \cdots, j_M)$ sources

$R_J + \sum_{i=1}^{M} r_i j_i$ denotes the rate corresponding to the collection J.

$\kappa_J = \sum_{i=1}^{M} \beta_i j_i$ the decay rate in set J.

$J_0 = argmin_J\{\kappa_J : R_J > (C - \rho)\}$

Lemma 3.1 *Consider a $(M/G/\infty)^M$ queueing system with M types of heterogeneous long-tailed inputs. Assume that $R_J \neq C - \rho$ for all J and with $C > \rho$. Then as $z \to \infty$*

$$\mathrm{P}[W_0 > z] = O(z^{-K_{J_0}}) \tag{3.1}$$

where W_0 denotes the stationary workload in the queue.

The above lemma gives us an estimate for the order of the tail of the workload in a single queue accessed by long-tailed inputs. The next lemma provides us with an estimate for the tail of the time the workload in a single queue fed only by light sessions is below the service rate.

Lemma 3.2 *Consider a single queue fed by long-tailed sources Y_t of the type described in the previous section with $E[Y_t] = \rho$. Recall that $Y_t = Y_t^L(\epsilon) + Y_t^H(\epsilon)$ for any $\epsilon > 0$. Let us further assume that $E[Y_t^L(\epsilon)] > C$ for some given $C \in [0, \rho)$. Then for any given $\gamma > 0$ the following holds true as $z \to \infty$ that,*

$$\mathrm{P}[\inf_{t \geq 0}\{X^L(-t, 0) - Ct\} > -\gamma z] \geq 1 - e^{-O(z)} \,. \tag{3.2}$$

Remark 3.1 *The above lemma is also valid if instead of long-tailed processes we have exponentially tailed ones i. e.*

$$\mathrm{P}[\tau_i > z] \sim e^{-O(z)} \text{ as } z \to \infty \,. \tag{3.3}$$

Finally we need the following result:

Lemma 3.3 *Consider a queueing $(M/G/\infty)^M$ system with $C > \rho$. Then for $Y_t^H(\epsilon)$ as defined in section 2 and for $\alpha > \epsilon$ as $z \to \infty$ the following hold true,*

$$a)\ \mathrm{P}[A_J(0)] \sim O(z^{-K_J}) \tag{3.4}$$

$$b)\ P[\text{All } J \text{ sessions active at 0 arrived before } -\alpha z] \sim O\Big(P[A_J(0)]\Big) = O(z^{-K_J}) \,. \tag{3.5}$$

4 N-queue long-tailed $H.O.L.$ system

We now state and provide an outline of the proof of the main result in this paper.

Theorem 4.1 *Consider an N-queue H.O.L.-scheduled queueing system with long-tailed $(M_i/G_i/\infty)$; $i = 1, 2, \cdots, N$ inputs as defined in Section 2. Assume that $R_{J_{1,2,..,k}} \neq C - \sum_{i=1}^{k} \rho_i$ for all $J_{1,2,..,k}$. Then for the k^{th} priority queue the overflow probability distribution for the stationary workload as $z \to \infty$ is given by:*

$$\mathrm{P}[W_{0,k} > z] = O(z^{-K_{J^0_{1,2,\dots,k}}}) \tag{4.1}$$

where

$$K_{J^0_{1,2,\dots,k}} := \min_{J_{1,2\dots,k}} \{K_{J_{1,2,\dots,k}} \;:\; R_{J_{1,2,\dots,k}} - (C - \sum_{i=1}^{k} \rho_i) > 0\} .$$

Proof We will prove that the probability under consideration is bounded above and below by quantities of the required order. Let us define,

$$A_{J_{1,\dots,k}}(-\alpha, 0) = \{\ A_{J_{1,\dots,k}}(0) \text{ and all heavy sessions active at time 0 arrived before } -\alpha z\} .$$

Upper bound

It is trivially seen that $W_{t,k} \leq \sum_{i=1}^{k} W_{t,i}$. Hence, invoking lemma 3.1 we obtain

$$P[W_{0,k} > z] \leq P[\sum_{i=1}^{k} W_{0,i} > z] \sim O(z^{-K_{J^0_{1,2,\dots,k}}}) \tag{4.2}$$

Lower bound

This part of the proof is based on the existence of a moment, say $-\alpha_2 z$, at which the first $k-1$ queues have in aggregate, sufficiently large amount of workload to guarantee that for the next $\alpha_2 z$ slots of time the k^{th} queue will not receive any service at all. Then we will show that for suitable choice of α_2 k^{th} queue will have at time 0 workload at least as much as z which proves our theorem. The following lemma provides us with a lower bound for the magnitude of α_2.

Lemma 4.1 *Consider a single queue fed by long-tailed sources Y_t of the type described in section 2, with $E[Y_t] = \rho$. Assume that $E[Y_t^L(\epsilon_1)] - C_1 > \Delta_1$ for some given $C_1 \in [0, \rho)$, $\Delta_1 \in (0, \rho - C_1)$. Then for any $\gamma_1 > 0$ and $\alpha > \frac{\gamma_1}{\Delta_1}$:*

$$\mathrm{P}[X^L(-\alpha z, 0) - C_1 \alpha z > \gamma_1 z] \geq 1 - e^{-O(z)} \qquad \text{as } z \to \infty . \tag{4.3}$$

Proof Let us define δ_1 such that $0 < \delta_1 < \Delta_1$. Such a δ_1 exists because Δ_1 is strictly positive. Then for $\gamma = \alpha\delta_1 - \gamma_1 > 0 (\Rightarrow \alpha > \frac{\gamma_1}{\delta_1} > \frac{\gamma_1}{\Delta_1})$ and $C = C_1 + \delta_1 < E[Y_t^L(\epsilon_1)]$ formula 3.2 applies yielding,

$$\begin{aligned}
1 - e^{-O(z)} &\leq P[\inf_{t \geq 0}\{X(-t, 0) - (C_1 + \delta_1)t\} > -(\alpha\delta_1 - \gamma_1)z] \\
&\leq P[X(-\alpha z, 0) - (C_1 + \delta_1)\alpha z > -(\alpha\delta_1 - \gamma_1)z] \\
&= P[X(-\alpha z, 0) - C_1 \alpha z > \gamma_1 z]
\end{aligned} \tag{4.4}$$

Which proves the lemma. ■

Proof of lowerbound

Let us first show that

$$P[\sum_{i=1}^{k} W_{-\alpha_2 z,i} > 2(1+C\alpha_2)z \; ; \; A_{J^0_{1,2,\ldots,k}}(-\alpha,0)] \sim P[A_{J^0_{1,2,\ldots,k}}(-\alpha,0)] \tag{4.5}$$

for some positive $\alpha = \alpha_1 + \alpha_2$ with $\alpha_1, \alpha_2 > 0$.

It follows from the definition $J^0_{1,2,\ldots,k}$ that in $(-(\alpha_1+\alpha_2)z, 0)$ work arrives to the subsystem consisting of the first k queues, at the rate $C - \sum_{i=1}^{k} \rho_i + \delta_2$, for some $\delta_2 > 0$ plus the arrival rate due to the light-$(1,2,\ldots,k)$ arrivals. Furthermore choose $\epsilon > 0$ such that

$$E[\sum_{i=1}^{k} Y^L_{t,i}(\epsilon)] - \sum_{i=1}^{k} \rho_i + \delta_2 > \Delta_2 \tag{4.6}$$

with $\Delta_2 \in (0, \delta_2)$ and

$$E[Y^L_{t,k}(\epsilon)] > \Delta \tag{4.7}$$

for some $\Delta \in (0, \rho_k)$.

Now for $\alpha_1 > \frac{2(1+C\alpha_2)}{\Delta_2}$ and any $\alpha_2 \geq 0$ one can see, applying the previous lemma with $C_1 = \sum_{i=1}^{k} \rho_i - \delta_2$ and $\gamma_1 = 2(1+C\alpha_2)$, that the first $\alpha_1 z$ slots of time suffice for the system of the first k queues to have at least $2(1+C\alpha_2)z$ of workload at the moment $-\alpha_2 z$. Indeed, invoking lemma 4.1,

$$\begin{aligned}
&P[\sum_{i=1}^{k} W_{-\alpha_2 z,i} > 2(1+C\alpha_2)z \; ; \; A_{J^0_{1,2,\ldots,k}}(-\alpha,0)] \\
&= P[\sum_{i=1}^{k} W_{-\alpha_2 z,i} > 2(1+C\alpha_2)z \mid A_{J^0_{1,2,\ldots,k}}(-\alpha,0)] \, P[A_{J^0_{1,2,\ldots,k}}(-\alpha,0)] \\
&\geq P[\sum_{i=1}^{k} X^L_i(-\alpha z, -\alpha_2 z) + (C - \sum_{i=1}^{k} \rho_i + \delta_2)\alpha_1 z - C\alpha_1 z > 2(1+C\alpha_2)z] P[A_{J^0_{1,.,k}}(-\alpha,0)] \\
&= P[\sum_{i=1}^{k} X^L_i(-\alpha z, -\alpha_2 z) - (\sum_{i=1}^{k} \rho_i - \delta_2)\alpha_1 z > 2(1+C\alpha_2)z] \, P[A^0_{J_{1,2,\ldots,k}}(-\alpha,0)] \\
&\geq (1 - e^{-O(z)}) \, P[A_{J^0_{1,2,\ldots,k}}(-\alpha,0)] \sim P[A_{J^0_{1,2,\ldots,k}}(-\alpha,0)]
\end{aligned}$$

and combining this with the trivial inequality,

$$P[\sum_{i=1}^{k} W_{-\alpha_2 z,i} > 2(1+C\alpha_2)z \; ; \; A_{J^0_{1,2,\ldots,k}}(-\alpha,0)] \leq P[A_{J^0_{1,2,\ldots,k}}(-\alpha,0)] \, ,$$

we obtain relation (4.5). Hence, (4.5) holds for any $A_J(-\alpha,0)$ with $J \geq J^0_{1,2,\ldots,k}$ with the convention that a vector is greater than (or equal) another one if all its co-ordinates satisfy this relation.

We will show that the event $\{A_{J^0_{1,2,\ldots,k}}(-\alpha,0)\}$ serves to compute an asymptotic lower bound for the order of the tail distribution of the k^{th} queue.

Recalling formula 3.5 we have

$$P[A_{J^0_{1,2,\ldots,k}}(-\alpha,0)] \sim O(z^{-K_{J^0_{1,2,\ldots,k}}}) \, . \tag{4.8}$$

Now for α as defined previously using (4.5) and the previous result we obtain,

$$
\begin{aligned}
O(z^{-K_{J^0_{1,2,\dots,k}}}) &\sim P[A_{J^0_{1,2,\dots,k}}(-\alpha,0)] \\
&\sim P[A_{J^0_{1,2,\dots,k}}(-\alpha,0)\ ;\ \sum_{i=1}^{k-1} W_{-\alpha_2 z,i} + W_{-\alpha_2 z,k} > 2(1+C\alpha_2)z] \\
&\leq P[A_{J^0_{1,2,\dots,k}}(-\alpha,0)\ ;\ \sum_{i=1}^{k-1} W_{-\alpha_2 z,i} > (1+C\alpha_2)z] + P[W_{-\alpha_2 z,k} > (1+C\alpha_2)z]
\end{aligned}
$$

Now, combining the fact that the total available service rate is C and that $\sum_{i=1}^{k-1} W_{-\alpha_2 z,i} > (1+C\alpha_2)z$ we infer that during $(-\alpha_2 z, 0)$ there always exists a non-empty buffer among the first $k-1$ implying that during this time interval the k^{th} queue does not receive any service at all. Defining $\alpha_2 > 1/\Delta$ and applying lemma 4.1 for $C_1 = 0$ and $\gamma_1 = 1$, we deduce that with probability $1 - e^{-O(z)}$ the content of buffer k at time 0 is greater than z. As far as the second term in the rhs of the last expression is concerned this trivially implies that at time 0 the k^{th} buffer will have backlogged workload greater than z (recall that the total service rate is C). Invoking lemma 4.1 we obtain,

$$
\begin{aligned}
&O(z^{-K_{J^0_{1,2,\dots,k}}}) \leq \\
&\quad \leq \ P[A_{J^0_{1,2,\dots,k}}(-\alpha,0)\ ;\ \sum_{i=1}^{k-1} W_{-\alpha_2 z,i} > (1+C\alpha_2)z] + P[W_{-\alpha_2 z,k} > (1+C\alpha_2)z] \\
&\quad \sim \ P[A^0_{J_{1,2,\dots,k}}(-\alpha,0)\ ;\ \sum_{i=1}^{k-1} W_{-\alpha_2 z,i} > (1+C\alpha_2)z]\ P[X_k^L(-\alpha_2 z,0) > z] \\
&\qquad\qquad + P[W_{-\alpha_2 z,k} > (1+C\alpha_2)z] \\
&\quad \sim \ P[W_{0,k} > z] + P[W_{0,k} > z] = O\Big(P[W_{0,k} > z]\Big)\ .
\end{aligned}
$$

Thus,

$$
P[W_{0,k} > z] \geq O(z^{-K_{J^0_{1,2,\dots,k}}})\ . \tag{4.9}
$$

The proof is now complete. ■

Theorem 4.1 can be easily generalized to the case when among the N queues comprising our model there exist queues which are fed by light-tailed (exponential for example) cf. 3.3 processes as long as the higher priority buffer has long-tailed inputs. We impose the latter restriction since otherwise the light-tailed inputs can be treated as in the case of standard fluid queues. We state this result without proof below.

Corollary 4.1 *Consider an N-queue H.O.L.-scheduled queueing system. Further, let $\mathcal{H} = \{1, i_1, \dots, i_H\}$ with $1 \leq H \leq N-1$, be a collection of $H+1$ queues out on the N contained in our system. Assume that all buffers in $\mathcal{H}$ are fed by long-tailed processes while the remaining buffers are accessed by exponentially tailed sources (3.3). Then as $z \to \infty$, the following holds true for the order of the $1 \leq k^{th} \leq N$ priority queue tail distribution,*

$$
\mathrm{P}[W_{0,k} > z] = O(z^{-K_{J^0_{1,i_1,\dots,i_k^{\max}}}}) \tag{4.10}
$$

where $i_k^{\max} = \max_{1\le m\le|\mathcal{H}|}\{i_m \ : \ i_m \le k\}$ *where* $|\mathcal{H}|$ *stands for the cardinality of the set* $\mathcal{H}$*. And*

$$K_{J^0_{1,i_1,\dots,i_k^{\max}}} := \min_{J^0_{1,i_1,\dots,i_k^{\max}}} \{K_{J^0_{1,i_1,\dots,i_k^{\max}}} \ : \ R_{J^0_{1,i_1,\dots,i_k^{\max}}} - (C - \sum_{i=1}^{k} \rho_i) > 0\} . \tag{4.11}$$

Furthermore, we assume that $R_{J^0_{1,i_1,\dots,i_k^{\max}}} \ne C - \sum_{i=1}^{k} \rho_i$ *for all* $J_{1,i_1,\dots,i_k^{\max}}$.

5 Implications of results

One immediate implication of the results is that light-tailed inputs must be given priority over long-tailed inputs since they do not affect the long-tail asymptotics except for the fact that the effective service rate available is reduced by the mean of the source.

Having the order of the tail distributions for each buffer in a queueing system with *H.O.L.* scheduler we can obtain an optimal ordering of the priorities knowing the rates r_i and decay constants β_i of the different sources based on minimizing the probability of buffer overflow for a given buffer size. For this objective using the result of theorem 4.1 we give the following algorithm. Let denote $\mathcal{N} = \{1, 2, \dots, N\}$.

```
    Optimal ordering
    $i_1 = argmax_{i\in\mathcal{N}}\{\beta_i\lceil\frac{C-\rho_i}{r_i}\rceil\}$
For k=2 to N do
    $i_k = argmax_{\{i\in\mathcal{N}/\{i_1,\dots,i_{k-1}\}\}}\{\beta_i j_i + \sum_{u=1}^{k-1}\beta_{i_u} j_{i_u} \ : \ r_i j_i + \sum_{u=1}^{k-1} r_{i_u} j_{i_u} > C - \rho_i - \sum_{u=1}^{k-1}\rho_{i_u}\}$
Endfor;
```

Now the order $i_1, i_2, \dots, i_N$ with a queue i_k having priority over all i_l with $k < l \le N$, is the one with the required property. There may exist more than one such orderings if any of the maximum above is achieved at more than one indices i. Then we should define some extra criteria for resolving these ties, e. g. billing based rules or delay based rule e. t. c.

Concluding remarks

The results obtained in this paper can readily be generalized to the case of each buffer being accessed by M_i types of heterogeneous long-tailed inputs much as in the single buffer case. Barring a few technical details all the proofs go through and will appear in [6].

References

[1] S. Borst, O. Boxma and P. Jelenkovic; *Generalized processor sharing with long-tailed traffic sources*, in Teletraffic Engineering in a Competitive World, P. Key and D. Smith eds., Elsevier, 1999, pp. 345-354.

[2] O. J. Boxma and V. Dumas; *Fluid queues with long-tailed activity period distributions*, Computer Communications, 1998, to appear.

[3] N. G. Duffield; *Economies of scale in queues with sources having power-law large deviation scalings*, Journal of Applied Probability, **33**, 1996, pp. 840-857.

[4] D. Black et al: *An architecture for differentiated services*, Internet draft, Internet Engineering Task Force (IETF), May 1998.

[5] P. R. Jelenkovic and A. A. Lazar; *Asymptotic results for multiplexing subexponential on-off sources*, Advances in Applied Probability, 1999, to appear.

[6] C. Kotopoulos, N. Likhanov and R. R. Mazumdar; *Buffer overflow asymptotics in systems with HOL service with heterogeneous long-tailed inputs*, in preparation.

[7] W. E. Leland, M.S. Taqqu, W. Willinger and D. V. Wilson; *On the self-similar nature of Ethernet traffic (extended version)*, IEEE/ACM Trans. on Networking, Vol. 2, No. 1, 1994, pp. 1-15.

[8] N. Likhanov and R. Mazumdar; *Cell loss asymptotics in buffers fed with a large number of independent stationary sources*, Journal of Applied Probability, **36** (1), 1999, pp. 86-96.

[9] N. Likhanov and R. Mazumdar; *Cell loss asymptotics in buffers fed with heterogeneous long-tailed sources*, pre-print June 1999, University of Essex, submitted for publication.

[10] Z. Liu, P. Nain, D. Towsley and Z-L. Zhang; *Asymptotic behavior of a multiplexer fed by long-range dependent process*, Journal of Applied Probability, **36** (1), 1999, pp. 105-118 .

[11] M. Parulekar and A. M. Makowski; *Tail probabilities for $M/G/\infty$ input processes*, Queueing Systems, 1999, to appear.

[12] V. Paxon and S. Floyd; *Wide area traffic: the failure of Poisson modeling*, IEEE/ACM Trans. on Networking, **3**, 1993, pp. 226-244.

[13] C. Rainer and R. R. Mazumdar; *A note on the conservation law for continuous reflected processes and its application to queues with fluid inputs*, Queueing Systems, **28**, 1998, pp. 283-291.

[14] S. Resnick and G. Samorodnitsky; *Activity periods of an infinite server queue and performance of certain heavytailed fluid queues*, pre-print ORIE, Cornell University, 1996

Scheduling and Control of Manufacturing Systems — a Fluid Approach*

Gideon Weiss

Department of Statistics
The University of Haifa
Mount Carmel 31905, Israel.
gweiss@stat.haifa.ac.il

Abstract

We suggest a fluid framework for solving scheduling and control problems of manufacturing systems. We formulate the fluid approximation, show how to solve it, give an important graphical display of the fluid solution, construct a schedule from the fluid solution, and provide probabilistic bound of its suboptimality.

1 Introduction

Scheduling and control of manufacturing systems has always been a very hard problem with insufficient tools to solve it. A compelling motivation for its further development is the semi-conductor wafer fabrication industry, which has the most costly and complex plants ever, and is very high tech and receptive to new ideas. The fluid approach to such problems which has been suggested in the past [1, 3, 19] has been shown to have strong theoretical justification [8, 15], and has recently gained much wider interest as a scheduling and control tool [12, 14, 4]. In this paper we give a complete framework for a fluid approach as the tool to schedule and control an entire industrial plant.

Scheduling and control of manufacturing systems can be modeled as a deterministic combinatorial optimization problem, or as a multiclass queueing network. Both formulations are of little practical help in trying to optimize these systems exactly. Furthermore, they both suffer from some conceptual shortcomings. We suggest a unified approach — consider finite horizon and on line policies.

The finite horizon system can be approximated by a multiclass fluid network. We discuss a conjecture that the fluid control problem is a relaxation of the manufacturing system problem under on-line control policies, and hence it provides a bound on the optimal objective.

The fluid control problem is an SCLP — separated continuous linear program. SCLP can be solved numerically [16, 5] and therefore are more tractable than combinatorial optimization or Markov decision problems. We present here a new, simplex-based finite

---

*This research is supported in part by US-Israel BSF grant 9400196 and by German-Israel GIF grant I-564-246/06/97

algorithm for SCLP with linear data, which extends greatly our understanding of these problems.

We advocate a new graphical display of the fluid solution. It displays the dynamics of the whole plant for the entire planning horizon, and we believe it is more informative than Gantt-charts or PERT-networks. It should give the manager a complete overview of his system, and how the fluid solution controls it. The information in the display can be used for hierarchical control of the plant: It provides clear overall manufacturing rates on a plantwide scale, which can be set as targets for various departments or manufacturing cells.

The fluid solution can also be used to provide a detailed heuristic schedule for the manufacturing system which emulates the fluid solution.

Furthermore, one can do a probabilistic analysis of this heuristic algorithm and find a probabilistic bound on the difference between the fluid and the heuristic objectives which sandwich the optimum. In some cases the difference for a size N problem is of order $\log N$.

A word of caution: While the above framework seems plausible and promising, this work is still in early stages. Hence the various theorems, models, and results are far from complete.

2 Job shops, queueing networks and multiclass fluid networks

The job-shop formulation of a manufacturing system consists of machines (manufacturing stations) $i = 1, \dots, I$, and routes (production processes) $r = 1, \dots, R$. Route r consists of ordered steps (r, k) where $k = 1, \dots, K_r$, which is carried out by machine $\sigma(r, k)$. C_i denotes the set of steps performed on machine i.

In the job shop scheduling problem one has a set of N jobs partitioned into $\mathcal{N}_1, \dots, \mathcal{N}_R$, where job $j \in \mathcal{N}_r$ has arrival (release) time $A(j)$, it moves on route r, and requires processing times $X_{r,k}(j)$. We let $m_{r,k}$ denote the average of $X_{r,k}(j)$ over $\mathcal{N}_r$.

A schedule for these jobs uses all this data. It determines job-step start and completion times $s_{r,k}(j), t_{r,k}(j)$, and departure times $D(j) = t_{r,K_r}(j)$. These are subject to one to one assignment of jobs to machines, order of steps, release times and processing times. The objective value of a schedule is some nondecreasing function of the $D(j)$. Because the number of candidate schedules (given by the sequence and early start) is finite, the optimal solution is well defined and can be found through combinatorial optimization. However, the problem is *NP*-hard and only very small examples can be solved. Hence, one resorts to heuristics to produce a suboptimal schedule.

We believe that the optimal schedule, as well as some of the heuristic schedules, are conceptually wrong, in that they place equal weight on current and future decisions, while in reality the data for future decisions is very inaccurate.

The multiclass queueing network formulation has less information about individual jobs (manufactured parts) but one assumes that jobs can be classified into groups with some common properties. Individual jobs are then randomly chosen from these common job types. In particular one often assumes renewal arrival times $A(j)$. Routing is more flexible, processing steps are listed as classes $k = 1, \dots, K$, and a job that leaves class k moves to class l with probability $P_{k,l}$. Processing times $X_k(j)$ are assumed i.i.d. from some distributions with mean processing times m_k.

The problems which one tries to solve are performance evaluation of some given stationary policies (including the question of stability), and finding an optimal steady state policy. The latter is a discrete state space Markov decision problem for memoryless interarrivals and processing times. More generally the state includes some elapsed times. While such MDPs can be solved in theory, it is again the unfortunate fact that only the tiniest most trivial problems can be solved in practice. Heuristic approaches include the use of diffusion approximations [11, 13, 18].

We believe that the emphasis on stationary policies and steady state performance measures is again conceptually wrong. In real production systems the initial system state is important, solution is sought for a finite time horizon, and the problem data is not stationary beyond the planning horizon or even within it.

Drawing on the common features of the deterministic and the stochastic models (of the same real system), we will look for *online policies* which for decision at time t use only $Q_k(t)$, the queue length or buffer content in class or buffer k. Not even the past history of $Q_k(\cdot)$ is available to the scheduler at t. Additional information will consist of the average arrival, routing, and service rates. Furthermore, we will deal with finite horizon problems: We will only schedule the initial jobs $Q_k(0)$ and the following arrivals up to a total of N jobs. We will consider two objective functions: Makespan — completion time of the last job, and weighted flowtime — integral of weighted buffer contents (holding costs) over time.

Posing the problem in this way the distinction between deterministic and stochastic models vanishes: If we solve the problem for a particular data instance, then the best online solution can be compared to the (better) optimal combinatorial solution, as well as to any (worse) heuristic solution. If we average this comparison over the population of instances indicated by the stochastic model we get the expected performance of the online problem, and we can compare it to the best combinatorial solution or to an online heuristic solution. Such a comparison is what is called a probabilistic analysis of an algorithm [7]. We will use fluid approximations to supply the heuristics and to help in the probabilistic analysis.

A multiclass fluid network consists of the same machines $i = 1, \dots, I$ and buffers $k = 1, \dots, K$, partitioned into C_i. However the buffer contents $x_k(t)$ are real numbers, and the dynamics of the system are deterministic and continuous: If machine $\sigma(k)$ devotes all its capacity to buffer k then fluid will move out of buffer k at constant rate $1/m_k$. A proportion $P_{k,l}$ of fluid leaving buffer k will move to buffer l. The capacity of each machine is infinitely divisible, and so fluid can be moved out of buffer k at any rate $0 < u_k(t) < 1/m_k$. The fluid control problem is to choose pumping rates (the controls) out of the various buffers, subject to (here e denote a vector of 1's, $M_{i,k} = m_k$ if $k \in C_i$, a is the vector of initial buffer contents, and α the input rates):

$$\begin{aligned}
&x(t) = a + \alpha t - \int_0^t (I - P')u(s)ds, \\
&Mu(t) \le e, \\
&u(t), x(t) \ge 0.
\end{aligned}$$

The objectives are (here 1 is the indicator function, $w \ge 0$ is the vector of weights, and the infinite horizon assumes that the system can be emptied in a finite time):

$$\text{Makespan} \qquad \min \int_0^\infty 1_{x(t) \neq 0} dt$$

$$\text{Weighted Flowtime} \qquad \min \int_0^\infty w' x(t) dt$$

More generally we will solve the problem over a finite horizon; substituting and integrating the second objective by parts we get the *fluid control problem* — find measurable $u(t)$ to

$$\begin{aligned}
&\text{Makespan} && \min \int_0^\infty 1_{x(t) \neq 0} dt \\
&\text{Weighted Flowtime} && \max \int_0^T (T-t) w'(I-P')u(t)dt \\
&\text{s.t.} && \int_0^t (I-P')u(s)ds \leq a + \alpha t, \qquad (1) \\
& && Mu(t) \leq e, \\
& && u(t) \geq 0, \quad 0 < t < T.
\end{aligned}$$

3 The Fluid Provides a Lower Bound

The multiclass fluid network can be obtained from a multiclass queueing network as a limit of fluid scalings: Let $n = |Q(0)|$ denote the norm (total number) of the initial state, then $x(t) = \lim_{n\to\infty} \frac{1}{n} Q(nt)$. It is by now quite well understood that stability of a multiclass queueing network depends on these fluid limit models [8]. It is also known now that the optimal solution of the fluid control problem (1) is closely related to MDPs of queueing networks [15].

The fluid formulation can also be obtained from the finite deterministic job-shop scheduling problem, through relaxation of the constraints in three steps:

Relax integrality of buffers: $Q_{r,k}(t)$ need not be integer.

Relax machine integrality: Machine is infinitely divisible and can work simultaneously on any number of job-steps.

Relax integrality of job-steps: Processing of the following job-step can start on the completed part of a current job-step.

Consider a job-shop scheduling problem with weighted flowtime objective:

$$\begin{aligned}
&\min \sum_{r=1}^{R} \sum_{j\in\mathcal{N}_r} \sum_{k=1}^{K_r} w_{r,k} \left(\frac{s_{r,k}(j) + t_{r,k}(j)}{2} - \frac{s_{r,k-1}(j) + t_{r,k-1}(j)}{2} \right) \\
&s_{r,0}(j) = t_{r,0}(j) = 0 \\
&t_{r,k}(j) = s_{r,k}(j) + X_{r,k}(j) \\
&s_{r,k}(j) \geq t_{r,k-1}(j) \qquad (2) \\
&I_{r,k}(j,t) \in \{0,1\} \\
&I_{r,k}(j,t) = 1 \Leftrightarrow s_{r,k}(j) \leq t < t_{r,k}(j) \\
&I_{r,k}(t) = \sum_{j\in\mathcal{N}_r} I_{r,k}(j,t) \\
&\sum_{(r,k)\in C_i} I_{r,k}(t) \leq 1
\end{aligned}$$

This is the so called time indexed formulation, and $I_{r,k}(j,t)$ indicates assignment of the machine $\sigma(r,k)$ to job-step (r,k) of job j at time t. The objective uses the so called half completion times, a job is counted as moving out of the buffer of step (r,k) halfway through its processing. The first constraint defines $s_{r,0}(j) = t_{r,0}(j) = 0$ which corresponds to all jobs being released at time 0, and no further arrivals.

We consider solving this problem by using an online policy. In that case, when a job-step is started we do not know its processing time $X_{r,k}(j)$. We also assume that no past records of the schedule are available, so that we cannot predict future processing times, except from their means $m_{r,k} = \frac{1}{|\mathcal{N}_r|}\sum_{j\in\mathcal{N}_r} X_{r,k}(j)$. One model for this is that the jobs are presented in a fixed order (time stamped), which has to be followed. In that case, when a machine becomes available the decision is just to choose which job-step to do, based only on the number of jobs waiting at each job-step. One would then want to look at the average performance over all the permutations of the processing times of $j \in \mathcal{N}_r$. Another model is to think of processing times as generated from a distribution with mean $m_{r,k}$ and again one would be interested in the average performance. Hence we look at the expected objective:

$$E\left\{\min \sum_{r=1}^{R}\sum_{j\in\mathcal{N}_r}\sum_{k=1}^{K_r} w_{r,k}\left(\frac{s_{r,k}(j)+t_{r,k}(j)}{2} - \frac{s_{r,k-1}(j)+t_{r,k-1}(j)}{2}\right)\right\} \tag{3}$$

under the above constraints. Here the constraints need to hold pathwise, and the policy is on line.

The fluid control problem corresponding to problem (2) is to choose $u_{r,k}(t), q_{r,k}(t)$ to:

$$\begin{array}{lll} \min \sum_{(r,k)} \int_0^\infty w_{r,k} q_{r,k}(t)dt & & \\ \text{s.t.} \quad \begin{array}{l} \int_0^t u_{r,1}(s)ds + q_{r,1}(t) = \mathcal{N}_r \\ \int_0^t (u_{r,k}(s) - u_{r,k-1}(s))ds + q_{r,k}(t) = 0 \end{array} & r = 1,\dots,R & \\ \sum_{(r,k)\in C_i} m_{r,k}u_{r,k}(t) \le 1, \quad i = 1,\dots,I & & \end{array} \tag{4}$$

Conjecture 3.1 *The solution of problem (4) is a lower bound on the expected value (3) of problem (2) under online scheduling policies.*

It was noted already in [19], and later in [9] that the fluid solution to the makespan problem is a lower bound for the job-shop problem, under any policy.

4 A Simplex Like Algorithm for Solution of Separated Continuous Linear Programs

Problem (1) is a special case of a separated continuous linear program; for history of SCLP see [2]. The SCLP problem (with linear data) is:

$$\begin{array}{lllll} & \max & \int_0^T (T-t)c'u(t)dt & & \\ & \text{s.t.} & \int_0^t Gu(s)ds \le a + \alpha t, & \text{slack } x(t), & \text{dual } \pi(T-t), \\ (SCLP) & & Hu(t) \le b, & \text{slack } z(t), & \text{dual } r(T-t), \\ & & u(t) \ge 0, \quad 0 < t < T & & \end{array}$$

where G is a $K \times J$ matrix, and H is an $I \times J$ matrix. Pullan has formulated a dual problem:

$$\begin{array}{llll} & \min & \int_0^T (a' + (T-t)\alpha')d\pi(t) + \int_0^T b'r(t)dt & \\ (DCLP) & \text{s.t.} & G'\pi(t) + H'r(t) \ge ct, & \text{slack } q(t), \quad \text{dual } u(T-t), \\ & & r(t) \ge 0, \pi(0) = 0, \pi \text{ non-decreasing}, \quad 0 < t < T & \end{array}$$

and has shown strong duality between SCLP and DCLP [17]. He also defined complementary slackness by:

$$\begin{aligned} x_k(t) > 0 &\Leftrightarrow \pi_k \text{ constant in neighborhood of } T-t \\ z_i(t) > 0 &\Leftrightarrow r_i(T-t) = 0 \\ u_j(t) > 0 &\Leftrightarrow q_j(T-t) = 0 \end{aligned}$$

It follows directly from weak duality that any pair of primal and dual complementary slack feasible solutions are optimal.

Associated with SCLP and DCLP is the following family of primal and dual linear programs, obtained by differentiation:

$$(LP)\quad \begin{array}{ll} \max & c'u \\ \text{s.t.} & Gu + \xi = \alpha, \\ & Hu + z = b, \\ & u \geq 0, z \geq 0 \end{array} \qquad (DP)\quad \begin{array}{ll} \min & \alpha'\dot{\pi} + b'\dot{r} \\ \text{s.t.} & G'\dot{\pi} + H'\dot{r} - \theta = c, \\ & \dot{\pi} \geq 0, \end{array}$$

Members of this family of linear programs and their duals differ in the sign constraints on the primal variables ξ (which are the time derivatives of $x(t)$) and the dual variables $\dot{r}$ and θ (which are the time derivatives of $r(t)$ and $q(t)$).

Pullan has shown that the optimal solution of SCLP is given by piecewise constant $u(t)$, with a finite bounded number of breakpoints. We show that a much stronger characterization exists:

Theorem 4.1 *Assume SCLP is feasible and bounded. Then there exists an optimal solution with piecewise constant $u(t)$ and breakpoints $0 = t_0 \leq t_1 \cdots \leq t_N = T$ of the following form: Let $\xi, \theta, \dot{r}, \dot{\pi}$ be the piecewise constant time derivatives of x, q, r, π respectively. Then in each interval (t_{n-1}, t_n) the values $u, z, \xi, \theta, \dot{r}, \dot{\pi}$ are complementary slack basic solutions of LP,DP which are optimal for the sign constraints*

$$\begin{array}{ll} \xi_k \geq 0 \textit{ if } x_k(t_{n-1}) = 0, & \xi_k \textit{ unrestricted if } x_k(t_{n-1}) > 0. \\ \theta_j \geq 0 \textit{ if } q_j(t_n) = 0, & \theta_j \textit{ unrestricted if } q_j(t_n) > 0. \\ \dot{r}_i \geq 0 \textit{ if } r_i(t_n) = 0, & \dot{r}_i \textit{ unrestricted if } r_i(t_n) > 0. \end{array}$$

Furthermore, these basic solutions in neighboring intervals differ only by a single pivot, and the breakpoint time t_n is determined by the pivot, through the linear equation:

$$\begin{array}{l} \textit{if } \xi_k \textit{ leaves the basis at } t_n \textit{ then } x_k(t_n) = 0 \\ \textit{if } u_j \textit{ leaves the basis at } t_n \textit{ then } q_j(t_n) = 0 \\ \textit{if } z_i \textit{ leaves the basis at } t_n \textit{ then } r_i(t_n) = 0 \end{array}$$

Proof. This theorem follows from the verification of the following algorithm. ∎

We shall now make the following non-degeneracy assumption: All basic solutions of LP and DP are non-degenerate. If G and H are full rank, this can always be achieved by perturbing α, b and c. Some further perturbation of T will assure that $0 = t_0 < t_1 \cdots < t_N = T$.

Given a solution with the above properties, we can write a set of equations for the interval lengths between the breakpoints, $\tau_n = t_n - t_{n-1}$. We get $N-1$ equations from the $N-1$ pivots at $t_1, \ldots, t_{N-1}$ and an additional equation from $t_N = T$. We also add the conditions:

$$\begin{array}{l} x_k(0) = a_k, \\ \pi_k(0) = 0, \quad r_i(0) = 0, \quad q_j(0) = 0. \end{array}$$

Such a solution is optimal if:

All $\tau_n > 0$.

The values of all q, r, x at all the points t_n are ≥ 0; these conditions need only be verified at t_n which are local minima of q, r, x.

Given an optimal solution, the solution will be optimal for a range of values of T. This range can be expressed by writing:

$$A\tau = d + e(T - \underline{T}), \qquad \text{solved by } \tau = \delta + \epsilon(T - \underline{T})$$
$$B\tau \geq g + h(T - \underline{T}), \qquad \text{solved by slack} = \gamma + \eta(T - \underline{T})$$

where A are the coefficients of the equations for the τ, and B are the coefficients for checking that the required values of q, r, x are non-negative. Assume: all components of δ and γ are non-negative, with some components equal to zero, and for the zero components the corresponding components of ϵ and η are positive. Define:

$$\overline{T} = \underline{T} + \min\{\frac{\delta}{-\epsilon}, \frac{\gamma}{-\eta} : \text{over components with }, \epsilon, \eta < 0\}$$

then the solution will be optimal exactly for all values of T in the interval $[\underline{T}, \overline{T}]$.

We now outline an algorithm to solve SCLP. The algorithm constructs at one and the same time the optimal solutions for all values of $0 < T < \infty$.

Step 1: Solution for small T: Solve LP for all $\theta \geq 0, \dot{r} \geq 0$, all ξ unrestricted. The resulting basic solution and its dual are optimal for small T.

Step 2: Given a solution of the form in the theorem, optimal for some $T \geq \underline{T}$, construct the equations and the inequalities, find $\overline{T}$ and locate its arg min. There are several possibilities:

If $\overline{T} = \infty$, the solution is optimal for all $T > \underline{T}$. Stop.

If arg min is obtained when $\tau_n = 0$ for some n, remove the intervals which have shrunk to zero. There will now be two events scheduled for the time $t_{n-1} = t_n$. Go to Step 3 to introduce a new interval, in which the event which was later is now the earlier event.

If arg min is obtained when at some t_n a previously positive x, r or q hit zero, go to Step 3 to introduce a new interval with the event that this x, r, q hit zero.

Step 3: To introduce a new interval, impose the appropriate sign constraints and solve the resulting LP problem. If the basis for the new interval is a single pivot apart from its neighboring intervals, go to Step 5. If the optimal solution of the new LP is more than one pivot apart from its neighbors go to Step 4.

Step 4: If the introduction of a new interval required more than one pivot, there needs to be a new interval for each of these additional pivots. These intervals are introduced in a sequence of steps which are similar to Step 2 and Step 3, except that the corresponding coefficients of ϵ, η are used instead of δ, γ, and coefficients from the new interval are used instead of ϵ, η, to determine the arg min in Step 2. This step can be nested.

Step 5: Update $\underline{T} := \overline{T}$, and the equations and inequalities $A\tau = d + e(T - \underline{T})$ and $B\tau \geq g + h(T - \underline{T})$. Return to Step 2.

Proof. It is necessary to show that the steps of the algorithm are doable, and that it does not cycle. We skip the details. ■

5 The Fluid Solution and its Graphical Representation

The fluid solution consists of piecewise constant flows, and piecewise linear continuous fluid levels in all the buffers. In this section we will denote the fluid levels by q, that is $q_k(t)$ or $q_{r,k}(t)$ as appropriate.

Consider first the case of a re-entrant line with initial fluid levels a and no input $\alpha = 0$. Let $q_k^+(t) = \sum_{j=1}^k q_k(t)$ be the cumulative amount of fluid that has not yet passed out of buffer k, i.e. the total fluid upstream of k. We plot all the $q_k^+(t)$ on a single graph as a function of the time t, and call this the *Fluid-Dynamics chart* of the system. At any time t it shows the column of fluid still in the system, divided into the various ordered buffers, from 1 (most upstream) to K. A horizontal cross section of the chart presents the motion of a particle of fluid, from the buffer in which it started initially, through all the buffers until it exits, from which one can read the sojourn times in the various buffers.

A similar picture can be drawn with inputs, except that the motion of each particle of fluid will now be along a longitudinal non-horizontal line.

For a system with several production processes, given by several routes, one can order all the production steps of all the routes and rename them as $k = 1, \ldots, K$, where the order of each route is preserved. This can be done in many ways, and one should chose one from familiarity with the actual plant. A plot of all the $q_k^+(t)$ will again present at any time the column of ordered fluid in the system. The longitudinal cross sections representing the motion of a particle of fluid from one of the routes, will have vertical jumps when it leaves a buffer, over buffers of other routes.

In any case the Fluid-Dynamics Chart presents an overall picture of how parts move through the system, and contains a wealth of information. If the display is computerized it can be enhanced by various colorings and different emphasis, presented in several layers. Figure 1 gives an example of Fluid Dynamics Charts.

6 Heuristic Implementation

The fluid solution can be used directly to provide a heuristic schedule. Let $q_k^+(t)$ be the fluid solution cumulative fluid levels, which we now take as the recommended solution. Let $Q_k^+(t) = \sum_{j=1}^k Q_k^{(}t)$ be the actual number of jobs which at time t have not yet passed beyond buffer k. Consider a decision moment t at which machine i becomes available. We can then apply it to the first (time stamped) job in buffer $k*$ defined by

$$k* = \arg\max \left\{ \frac{Q_k^+(t) - q_k^+(t)}{q_k^+(t)} : k \in C_i, k \text{ is non-empty} \right\}.$$

This means that we try to get close to the fluid solution by giving priority to the buffer in which the relative deviation from the fluid solution is maximal. Many other versions of this can be implemented.

Implementation of this or similar policies will often give a cyclic or near cyclic assignment of a machine to its various buffers. It will also tend to create safety stocks at the busiest buffers. Some experience with theses ideas is reported in [4, 6, 9]. We note that the policies obtained are in some case similar to the *drum buffer rope* policies of Goldratt [10].

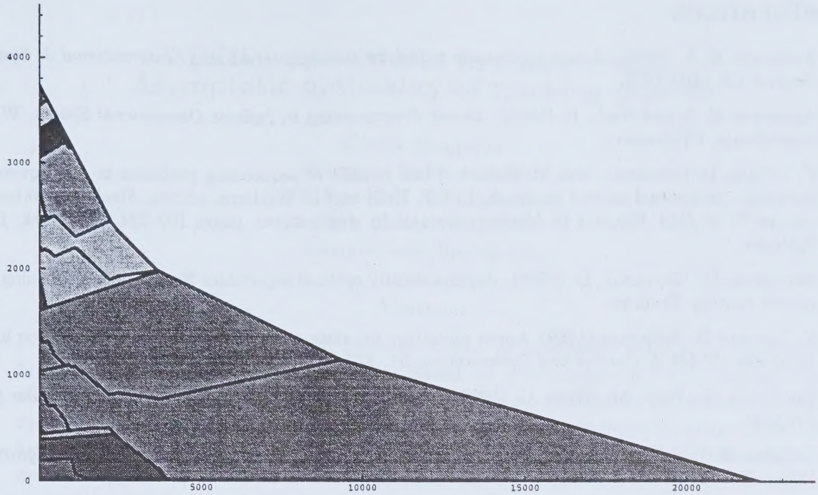


Figure 1: Fluid Chart of a 5 Machine 20 Buffer Re-Entrant Line

7 Probabilistic Analysis of the Heuristic and Probabilistic Performance Bounds

The classical job-shop scheduling problem is to minimize the makespan. In many practical situations there are many jobs on each route, which differ in their processing times. The fluid approach gives a complete treatment of this problem. First of all, the fluid solution (non-unique) is to use constant flow rates which will empty all buffers at one and the same time, and will keep at least one of the machines fully occupied throughout. It is trivial to see that the fluid solution provides a lower bound on the combinatorial job-shop makespan under any policy, not just online policies.

Dai and Weiss [9] propose a 'drum buffer rope' type heuristic which emulates this fluid solution. It starts by building safety stocks of jobs at all buffers, and then continues by letting the busiest (bottleneck) machine set the pace and other machines keep up with it. Dai and Weiss then perform a probabilistic analysis of their heuristic. They assume:

The number of jobs on each route is $\leq N$.

Processing times $X_{r,k}(j)$ are i.i.d. with mean $m_{r,k}$ and finite exponential moments for some $\theta > 0$, $E\{e^{\theta X_{r,k}(j)}\} < \infty$.

There is a unique bottleneck machine.

In that case the fluid makespan is of order $O(N)$, and:

Theorem 7.1 *There exist constants c', c such that the fluid heuristic with safety stocks $S = \lceil c'(\log N + 1) \rceil$ satisfies:*

$$P\{\text{Heuristic makespan } - \text{Fluid makespan } > c(\log N + 1)\} < \frac{1}{N}.$$

References

[1] Anderson, E. J. (1981). A new continuous model for job-shop scheduling. *International J. Systems Science* **12**, 1469–1475.

[2] Anderson, E. J. and Nash, P. (1987). *Linear Programming in Infinite Dimensional Spaces.* Wiley-Interscience, Chichester.

[3] F. Avram, D. Bertsimas, and M. Ricard. Fluid models of sequencing problems in open queueing networks: an optimal control approach. In F.P. Kelly and R. Williams, editors, *Stochastic Networks*, volume 71 of *IMA Volumes in Mathematics and its Applications*, pages 199–234, New York, 1995. Springer.

[4] Bertsimas, D., Gamarnik, D. (1998). Asymptotically optimal algorithms for job shop scheduling and packet routing. Preprint.

[5] X. Luo and D. Bertsimas (1998) A new algorithm for state constrained separated continuous linear programs. *SIAM J. Control and Optimization*, **37**, 177–210. To appear.

[6] Boudoukh, T., Penn, M., Weiss, G. (1998) Scheduling jobshops with some identical or similar jobs. Preprint .

[7] Coffman, E.G. and Lueker, G.S. (1991) *Probabilistic Analysis of Packing and Partitioning Algorithms* Wiley, New York.

[8] Dai, J.G. 1995. On positive Harris recurrence of multi-class queueing networks: a unified approach via fluid limit models. *Annals of Applied Probability* **5**, 49–77.

[9] J. B. Dai and G. Weiss (1999). A Fluid Heuristic for minimizing Makespan in Job-Shops. Preprint.

[10] Goldratt, E.M. and Cox, J. *The Goal: A Process of Ongoing Improvement.* 2nd edition North River Press, 1992.

[11] Harrison, J.M., 1988. Brownian models of queueing networks with heterogeneous customer populations. *Proceedings of the IMA Workshop on Stochastic Differential Systems*, Fleming W., Lions P.L., editors, Springer-Verlag.

[12] Harrison, J.M., (1996). The BIGSTEP approach to flow management in stochastic processing networks, *Stochastic Networks: Theory and Applications*, F. P. Kelly, I. Ziedins, S. Zachary, Editors, Oxford University Press.

[13] Kushner, H.J. and Ramachandran, K.M. (1989) Optimal and approximately optimal control policies for queues in heavy traffic. *SIAM J Control and Optimization* **27**, 1293–1318.

[14] Magalaras, C. (1998) Discrete-review policies for scheduling stochastic networks: fluid asymptotic optimality. Submitted to *Annals of Applied Probability.*

[15] Meyn, S.P 1995 The policy improvement algorithm for Markov decision processes with general state space. *IEEE Transactions on Automatic Control*, to appear.

[16] Pullan, M. C. (1993). An algorithm for a class of continuous linear programs. *SIAM J. Control and Optimization* **31**, 1558–1577.

[17] Pullan, M. C. (1996). Existence and duality theory for separated continuous linear programs. *Mathematical Modeling of Systems* **1** to appear.

[18] Wein, L.M., (1992). Scheduling networks of queues: heavy traffic analysis of a multistation network with controllable inputs. *Operations Research* **40**, S312–S334.

[19] Weiss, G. (1995) On optimal draining of fluid re-entrant lines. In *Stochastic Networks — IMA Volumes in Mathematics and its Applications*, Editors: F.P. Kelly and Ruth Williams, Springer-Verlag, New York, 93–105.

Multiclass networks in heavy traffic: Asymptotic optimality of tracking policies

Costis Maglaras
Columbia Business School
409 Uris Hall, 3022 Broadway
New York, NY 10027
c.maglaras@columbia.edu

Abstract

This paper is concerned with optimal control of stochastic processing networks. Specifically, the approach taken here is based on approximating (or replacing) the stochastic network by its Brownian analog, solving an associated Brownian optimal control problem. and then using the derived Brownian control policy in order to define an implementable rule for the stochastic network. Using an illustrative example, we describe a general mechanism that translates the Brownian control law to an implementable control policy in the stochastic network that is asymptotically optimal in the heavy traffic limiting regime, where the Brownian model approximation becomes valid. The proposed method is a receding horizon tracking policy that continuously allocates resources in order to track a desired target state that is derived from the solution of the Brownian control problem. Implementation of such policies involves the enforcement of small safety stocks.

1 Introduction

This paper is concerned with optimal control for open multiclass queueing networks. In these networks there are many job classes that may differ in their arrival processes, service requirements, and routes through the network, and there is a many-to-one relation between job classes and servers. Decisions are to be made as to the sequence in which jobs of various classes are served at each station, and a scheduling policy is a rule according to which these sequencing decisions are made. These models arise in many application areas such as manufacturing networks, service operations, packet switched communication networks, and multiprocessor computer systems.

Dynamic control problems for stochastic processing networks are both analytically and computationally hard. While most often one relies on the use of heuristics that are validated through simulation studies, one approach that has emerged from research over the past 10-15 years is based on a hierarchy of approximate models that provide tractable "relaxations" of these problems as a framework for analysis and synthesis. In particular, the analytical theory associated with fluid and diffusion approximations has produced important insights in understanding how the performance of a multiclass network depends on different design and control parameters. This is our starting point. Specifically, the approach taken here is based on approximating (or replacing) the stochastic network by its Brownian analog, solving an associated Brownian optimal control problem, and then using the derived Brownian control policy in order to define an implementable rule for the stochastic network. Finally, the proposed implementation is tested for asymptotic optimality in the heavy traffic limiting regime, where the Brownian model approximation becomes valid. In comparison to the original problem at hand, this criterion provides a relaxed notion of optimality that appears to be more tractable, and it provides a framework for analysis and synthesis of near-optimal control policies.

This framework was proposed by Harrison in [11], and has been successfully applied by many researchers in specific examples such as [15, 16, 19, 22]. However, a general theory, as it was envisioned in [11] and summarized above, is still beyond our reach due to the following reasons. First, the Brownian control problems, while vastly simpler than the original problem at hand, are still quite intricate: only a small subset of them admit closed form solutions, and the numerical techniques developed in [18], although a major breakthrough in this area, they are still only applicable to small scale problems. Second, the Brownian control policy does not have a direct interpretation in the original stochastic network, and proposed translations in the examples that appear in the papers mentioned above have exploited specific problem structure and are hard to generalize. Finally, the heavy traffic asymptotic analysis of the implemented policies had proved to be very complex even for simple examples [19, 22]. A discussion of these observations and a comprehensive literature review can be found in [25].

A series of three recent developments, however, have provided a catalyst for important new results in this area. The first is the theory of fluid models developed in [6, 4, 8, 7, 2] that has been very successful in stability analysis for multiclass networks, and in the synthesis of good scheduling policies for network control problems; see [5, 1, 23]. The second is the description of a general policy translation mechanism, named BIGSTEP, by Harrison in [13]: this mechanism reviews system status in discrete time intervals and makes scheduling decisions over each of these intervals using the solution of the Brownian control problem. Following the controller synthesis framework described above but using fluid instead of Brownian approximations, the author generalized the BIGSTEP ideas in [21, 20] to form a family of trajectory tracking discrete-review policies that were shown to be asymptotically optimal in the fluid limiting regime (which is related to transient optimization). That is, we have a technique to translate fluid model control policies, which are transient trajectories starting from arbitrary initial conditions, into implementable scheduling rules that achieve the desired fluid limiting response. Finally, the third is the development of a powerful new technique by Bramson [3] and Williams [27, 26] that renders the diffusion scale asymptotic analysis into essentially a verification of fluid scale properties –which is much simpler. Their results were developed for networks under fixed scheduling policies -that is, without dynamic control capability. Kumar used this framework in [17] to prove asymptotic optimality of a heuristic policy, also a static priority, that was proposed by Harrison and Wein [11] based on the solution of a Brownian optimal control problem associated with two station closed networks.

In this paper, starting from the three points mentioned above, we will describe a general mechanism, in the form of a receding horizon tracking policy, that translates the solution of the approximating Brownian control problem into an implementable scheduling policy that is asymptotically optimal in the heavy traffic limit. Our solution draws useful insights from the Bramson-Williams results that connect the Brownian model to its "intermediate" fluid model, and then applies the trajectory tracking framework developed in [20] in order to achieve the "intermediate" fluid model response that yield the desired diffusion scale behavior. Most of the results presented here will be developed in the context of a simple two station multiclass network first studied by Harrison and Wein [15], and further analyzed by Martins, Shreve and Soner [22] and Kushner and Martins in [19]. In fact, our main result of heavy traffic asymptotic optimality has already been established for a variant of a different translation proposed in [15], and rigorously analyzed in a Markovian setting in [22]. However, the results in [15, 22] are hard to extend to more general problems. Their proposed policy

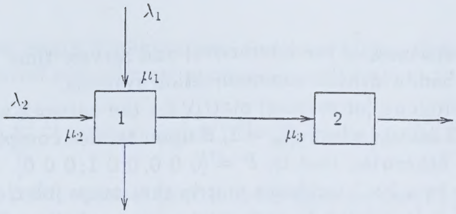


Figure 1: *Criss-Cross* network: $\lambda_1 = \lambda_2 = \rho,\ \mu_1 = \mu_2 = 2,\ \mu_3 = 1$

depends on the particular structure of this specific network, and their asymptotic analysis, based on viscosity solutions, is quite involved. In contrast, our solution will not depend on the peculiarities of the specific example under investigation, and our analysis, based on the Bramson-Williams framework, will be very simple and generalizable.

The remaining of the paper is structured as follows. §2 describes the criss-cross network under investigation, its Brownian analog, and the solution of the associated Brownian control problem. In §3 we describe the proposed translation mechanism and in §4 we state the main result of heavy traffic asymptotic optimality and provide an outline of its proof.

2 The criss-cross network and its Brownian analog

2.1 *The criss-cross queueing network*

We will consider the so called *criss-cross* network, first introduced in [15], which is shown in Figure 1. This is a multiclass network with three job classes. Class 1 jobs require service at station 1 and then they exit the network. while class 2 jobs require service first at station 1 and then at station 2. We associate infinite capacity buffers with each job class.

Jobs of classes 1 and 2 arrive according to independent renewal processes with rates λ_1 and λ_2 respectively. The interarrival time distributions have finite variances a_1^2 and a_2^2 respectively. Service times are i.i.d. positive random variables drawn from some general probability distribution specific to each job class. Mean processing times will be denoted by m_k and the corresponding processing rates by μ_k. The variance of the service time distributions will be denoted by $s_k^2 < \infty$, for $k = 1, 2, 3$. All processes are assumed to be mutually independent. Jobs within a class are served on First-In-First-Out (FIFO) basis and preemptive-resume type of service is assumed. We strengthen the probabilistic assumptions by requiring that the interarrival and service time distributions have finite exponential moments for a sufficiently small exponent; this restriction will be used for some large deviations estimates needed in our analysis. In particular, let $E_k(t)$ be the cumulative number of class k external arrivals up to time t for $k = 1, 2$, $V_k(n)$ be the total service requirement for the first n class k jobs for $k = 1, 2, 3$, and let $V_k(t) = V_k(\lfloor t \rfloor)$. The exponential moment and i.i.d. assumptions imply that for any class k, $\epsilon > 0$ and n sufficiently large,

$$\mathrm{P}\left(\sup_{0\leq s\leq n} |E_k(s) - \lambda_k s| > n\epsilon\right) \leq e^{-f_a^k(\epsilon)n} \text{ and } \mathrm{P}\left(\sup_{0\leq s\leq n} |V_k(s) - m_k s| > n\epsilon\right) \leq e^{-f_s^k(\epsilon)n}, \quad (1)$$

where the exponents $f_a^k(\epsilon), f_s^k(\epsilon) > 0$. Roughly speaking, it is required that a large deviations upper bound to hold for the sample paths of the processes $E_k(\cdot)$ and $V_k(\cdot)$. Upper bounds of the form of (1) follow from Mogulskii's Theorem [10, Thm 5.1.2], and a similar result exists for weakly dependent sequences [9]. The exponents in (1) can be taken as primitive data that describe the burstiness of the interarrival and service time processes; this is similar to the use of effective bandwidths in communication systems.

As usual, the switching (or routing) matrix for the network will be denoted by P, and is defined to be a 3×3 matrix where $P_{kl} = 1$, if upon service completion a class k job becomes a class l job, and 0 otherwise; that is, $P = [0\ 0\ 0; 0\ 0\ 1; 0\ 0\ 0]$. Similarly, the constituency matrix is defined to be a 2×3 incidence matrix that maps job classes to their corresponding servers as follows $C = [1\ 1\ 0; 0\ 0\ 1]$; each row corresponds to a different server.

The vector of effective arrival rates to each class, denoted by α, is given by $\alpha = [\lambda_1, \lambda_2, \lambda_2]'$, where $'$ denotes a transpose. The traffic intensity at each station is given by $\rho_1 = \alpha_1/\mu_1 + \alpha_2/\mu_2$ and $\rho_2 = \alpha_3/\mu_3$. For the parameter choices shown in the figure caption, we have that $\rho_1 = \rho_2 = \rho$; that is, the system is balanced. In vector form, $\alpha = (I - P')^{-1}\lambda$ and $\rho = CR^{-1}\lambda$, where $R = (I - P')M^{-1}$.

Denote by $Q_k(t)$ the total number of class k jobs in the system at time t, and by $Q(t)$ the corresponding 3-vector of "queue lengths." We define $|Q(t)| = \sum_k Q_k(t)$. A scheduling policy is a rule according to which server allocation decisions are made at any given time. It takes the form of a 3-dimensional cumulative allocation process $\{T^x(t),\ t \geq 0;\ T^x(0) = 0\}$, where $T_k^x(t)$ denotes the time allocated into serving class k jobs by the corresponding server up to time t, and the superscript "x" denotes the dependence on the initial condition. The cumulative allocation process should be non-decreasing, $T^x(0) = 0$, $CT^x(t) \leq t\mathbf{1}$ (where $\mathbf{1}$ denotes the vector of ones of appropriate dimension), and finally it must be non-anticipating; the last restriction implies that current allocations only depend on information available up to time t. Finally, the cumulative idleness process up to time t is given by $I(t) = t\mathbf{1} - CT(t)$.

The optimal control problem one starts with is the following: choose a cumulative allocation process $\{T^x(t), t \geq 0\}$ to minimize

$$\mathrm{E}\left(\int_0^\infty e^{-\gamma t} \sum_k Q_k(t)dt\right), \tag{2}$$

where $\gamma > 0$ is a positive discount factor. This control problem is very hard but, as we shall see next, its Brownian analog admits a very simple analytical solution under the specific choice of equal linear holding costs and service rates μ_k. On the other hand, the translation problem, which is the focus of this paper, is still subtle, and it provides a good first test for our proposed methodology. Moreover, as mentioned above, this is the case studied in [15] and rigorously analyzed in [22].

2.2 *The Brownian approximating model*

Let's focus now to the limiting case where $\rho \uparrow 1$, both servers approach a nominal load equal to 1, and the system is in the so called "heavy traffic" regime.

The long run nominal allocation for class k jobs is given by $\alpha_k t/\mu_k$ -this is the nominal effort required to keep up with the incoming traffic- and the corresponding 3-vector of nominal time allocations is given by $R^{-1}\lambda t$. When $\rho = 1$, $CR^{-1}\lambda = \mathbf{1}$; i.e., the system is operating at full capacity. Let $\delta(t) = R^{-1}\lambda t - T(t)$ be the deviation control that measures how far is the control $T(t)$ from the vector nominal of time allocations. Note that $I(t) = C\delta(t)$. Let $D_k(t)$ be the number of service completions when the appropriate server has allocated t time units in processing class k jobs. The queue length dynamics are given by

$$Q(t) = z + (E(t) - \lambda t) - (I - P')\left(D(T(t)) - M^{-1}T(t)\right) + \lambda t - RT(t), \tag{3}$$

where z is the initial queue length configuration. Let $n = 1/(1-\rho)$ and define the scaled queue length, deviation control and idleness processes as follows

$$Z^n(t) = \frac{Q^n(n^2t)}{n}, \quad U^n(t) = \frac{I^n(n^2t)}{n} \quad \text{and} \quad Y^n(t) = \frac{\delta^n(n^2t)}{n}. \tag{4}$$

Letting $\rho \uparrow 1$ and using the functional central limit theorem for (E, V), we can approximate the centered terms of cumulative arrivals and departures by a Brownian motion with appropriate drift and variance. Denoting by $Y(t)$ and $U(t)$ the limits of $Y^n(t)$ and $U^n(t)$ respectively, system dynamics can be approximated by the following set of equations

$$Z(t) = z + X(t) + RY(t), \quad X \sim (\zeta, \Sigma)\ BM, \tag{5}$$

$$U(t) = CY(t), \ U \text{ non-decreasing with } U(0) \geq 0, \tag{6}$$

$$Y \text{ is adapted to } X \text{ and } Z(t) \geq 0. \tag{7}$$

The associated Brownian optimal control problem will be to choose the control processes (Y, U) in order to minimize

$$\mathbf{E}\left(\int_0^\infty e^{-\gamma t} \sum_k Z_k(t)dt\right),$$

Let $W_1(t) = Z_1(t)/2 + Z_2(t)/2$ and $W_2(t) = Z_2(t) + Z_3(t)$. In vector form this can be rewritten as $W(t) = CR^{-1}Z(t)$, where $W_i(t)$ is the expected amount of work that server i has to exert in order to clear the current backlog in the system assuming there are no other external arrivals after time t. The workload formulation of the Brownian control problem is to choose RCLL (right continuous with left limits) processes Z and U to

$$\text{minimize} \quad \mathbf{E}\left(\int_0^\infty e^{-\gamma t} \sum_k Z_k(t)dt\right), \tag{8}$$

$$\text{subject to} \quad W(t) = W(0) + \xi(t) + U(t), \tag{9}$$

$$W(t) = CR^{-1}Z(t), \quad Z(t) \geq 0, \tag{10}$$

$$U \text{ non-decreasing with } U(0) \geq 0, \tag{11}$$

$$U \text{ and } Z \text{ are adapted to } \xi, \tag{12}$$

where $W(0) = CR^{-1}z$ and $\xi(t) = CR^{-1}X(t)$ for $X \sim (\zeta, \Sigma)\ BM$. The optimal Brownian control policy will not depend on (ζ, Σ), and for that reason we will not proceed here to characterize these parameters in terms of primitive problem data.

The reader is referred to [11] for an introduction to Brownian approximating models, and to [14] for a discussion of equivalent workload formulations.

2.3 *The Brownian optimal control policy*

As it was shown in [15, §3], the workload formulation (8)-(12) admits a surprisingly simple solution that is also *pathwise* optimal; i.e., it simultaneously minimizes $\sum_k Z_k(t)$ for all times t with probability one.

The structure of the optimal policy will have two ingredients: (i) we will choose the U process to control $W(t)$ through (9); and (ii) at any point in time we will choose the queue length configuration $Z(t)$ that holds the current workload $W(t)$, that is, $W(t) = CR^{-1}Z(t)$, and minimizes $\sum_k Z_k(t)$. Hence, as W changes due to random fluctuations of ξ and the control U, we will be instantaneously swapping Z's to hold the workload $W(t)$ while minimizing cost. From (5), instantaneous displacement between two vectors z_1, z_2 that hold the same workload, that is $CR^{-1}(z_1 - z_2) = 0$, can be achieved by applying the jump control $dY = R^{-1}(z_1 - z_2)$; this leaves the workload process unaffected since $dU = CdY = 0$.

We append a superscript * to denote the various processes under the following optimal Brownian control policy.

1. Keep $W^*(t)$ in convex region $S^* \equiv \mathbb{R}_+^2$;
2. $dU^* = 0$ in the interior of S^*.
3. Instantaneous "reflection" at ∂S^* using continuous and non-decreasing controls U_i^* and

$$U_i^* \text{ can increase only when } W_i^*(t) = 0 \implies \int_0^\infty W_i^*(t)dU_i^*(t) = 0, \text{ for } i = 1, 2; \quad (13)$$

   for $S^* = \mathbb{R}_+^2$, this implies that $U_i^*(t) = -\inf_{0\le s\le t} \xi_i(s)$, for $i = 1, 2$.
4. Given $W^*(t)$, choose $Z^*(t) = \Delta(W^*(t))$ where

$$\begin{aligned}\Delta(W^*(t)) &= \operatorname{argmin}\left\{\sum_k z_k : CR^{-1}z = W^*(t),\ z \ge 0\right\}\\ &= \operatorname{argmin}\left\{\sum_k z_k : z_1/2 + z_2/2 = W_1^*(t),\ z_2 + z_3 = W_2^*(t),\ z \ge 0\right\}. \quad (14)\end{aligned}$$

   Using the notation $[x]^+ = \max(0, x)$, $Z^*(t)$ is given by

$$Z_1^*(t) = [2W_1^*(t) - W_2^*(t)]^+,\ Z_2^*(t) = 2W_1^*(t) \wedge W_2^*(t),\ Z_3^*(t) = [W_2^*(t) - 2W_1^*(t)]^+. \quad (15)$$

For future use, we will denote by J^* the optimal cost $J^* \triangleq \mathrm{E}(\int_0^\infty e^{-\gamma t} \sum_k Z_k^*(t)dt)$.

In more complex Brownian control problems the region S^* and the directions of control on the boundary ∂S^* can be different, but the overall structure of the policy will remain the same. Note that using (14) one can express the objective in (8) in terms of the workload process as $\sum_k (\Delta(W(t))_k$, and essentially decouple steps (1)-(3) from step (4). The former will determine the idleness control U^*, while the latter will translate the W^* position to a queue length configuration of minimum cost.

3 Continuous-review policy translation

The main issues that need to be addressed by any successful translation of the Brownian optimal control policy are the following: (i) server idleness should only be incurred when server workload is on ∂S^*, which, for the criss-cross network, implies that server 2 can only idle when both classes 2 and 3 are empty; and (ii) given $W^*(t)$, the system should instantaneously move to $Z^*(t)$ which is the minimum cost position for that workload.

Exploiting the specific structure of the criss-cross network, Harrison and Wein proposed the following policy [15]: *server 1 gives priority to class 1 unless class 3 falls below some threshold θ, in which case class 2 gets priority.* To gain some intuition note that by giving priority to class 1 jobs we will be holding most of the system's workload in class 2 jobs -this is consistent with (15), and by reversing priorities below θ we will prevent server 2 from idling when there is work buffered at class 2. This is probably the simplest translation possible, and for the right choice of θ this policy will also be asymptotically optimal; a slight variation of this result was proved in [22] and it will also follow from our analysis, whereby the appropriate threshold will also be specified. However, while this proposition is ingenious, it is problem specific and hard to generalize outside the realm of this example.

To motivate our translation mechanism, we start by recognizing that the structure of the Brownian control policy is of the form: given $Z(t)$, pick a target state given by $\Delta(CR^{-1}Z(t))$ and go there instantaneously. This is summarized in the following schematic

$$Z(t) \longrightarrow W(t) = CR^{-1}Z(t) \longrightarrow \Delta(W(t)) = \Delta(CR^{-1}Z(t)).$$

To implement this instantaneous displacement in Brownian scale we will exploit the connections between fluid and Brownian models developed in [12] and [3]. First, recall that fluid scaled processes are defined by

$$q^n(t) = \frac{Q^n(nt)}{n}, \quad u^n(t) = \frac{I^n(nt)}{n} \quad \text{and} \quad y^n(t) = \frac{\delta^n(nt)}{n}, \tag{16}$$

and denote by q, u, y the fluid limits of these scaled quantities respectively.

Key observation: [3, Assmpt.3.1, Thm.4]: *Asymptotic tracking in fluid scale, that is,* $q(t) \to \Delta(CR^{-1}z) \triangleq q(\infty)$ *as* $t \to \infty$, *implies instantaneous displacement from* z *to* $q(\infty)$ *in Brownian time scale.*

This follows from the fact that the Brownian scaling in (4) involves an additional time compression factor of n over that embodied in (16). Hence, movements that take finite time in fluid scale appear as being instantaneous in Brownian scale.

Basic idea: This fact suggests the following *tracking* policy: at any given time, observe the queue length vector $Q(t)$, pick a target state $\Delta(CR^{-1}Q(t))$ -this is where we use the Brownian optimal control policy- and move to the target state in minimum time.

Minimum time control: The fluid model equations are given by

$$q(t) = z + Ry(t), \quad u(t) = Cy(t), \quad u(0) = 0, \quad \dot{T}(t) \geq 0, \quad C\dot{T} \leq \mathbf{1}, \tag{17}$$

where the last two constraints can be restated as $\dot{y}(t) \leq \alpha$ and $\dot{u}(t) \geq \mathbf{1}(1-\rho)$ respectively. Since $q(\infty) = z + Ry(\infty)$ it follows that $y(\infty) = R^{-1}(q(\infty) - z)$. Let t^* be the time it takes to reach $q(\infty)$. We have that $T(t^*) = \alpha t^* - R^{-1}(q(\infty) - z)$. Since, $\bar{T}(t^*) \geq 0$, it follows that

$$t^* \;\geq\; t^*(z, q(\infty)) \;\triangleq\; \max_k \frac{(R^{-1}(q(\infty) - z))_k}{\alpha_k}. \tag{18}$$

The capacity constraint is also satisfied since $CT(t^*) = \rho \mathbf{1} t^* - u(t^*) \leq t^* \mathbf{1}$. It is easy to verify that the minimum time control at any given time is to set allocation levels to

$$\beta(q(t), q(\infty)) \;\triangleq\; \frac{T(t^*(q(\infty), q(t))}{t^*(q(t), q(\infty))} \;=\; \alpha - \frac{R^{-1}(q(\infty) - q(t)))}{t^*(q(t), q(\infty))}. \tag{19}$$

The corresponding trajectory, denoted by $\Psi(\cdot; z, q(\infty))$, is simply translation along a straight line from $q(0) = z$ to its target $q(\infty) = \Delta(CR^{-1}z)$, and it is given by $q(t) = z + (q(\infty) - z)t/t^*(z, q(\infty))$, for $t \leq t^*(z, q(\infty))$ and $q(t) = q(\infty)$, for $t > t^*(z, q(\infty))$.

Safety stocks: To achieve the minimum time fluid trajectory defined through (19), we will enforce safety stock requirements that will prevent the queue length process from approaching the boundaries, where the network behavior becomes very different from its fluid idealization invoked above. Safety stocks were introduced in BIGSTEP [13], and have proved ubiquitous in achieving asymptotically exact tracking of target fluid trajectories in [20]. We enforce safety stock requirements of size $\theta(n)$, where $\theta : \mathbf{R}_+ \to \mathbf{R}_+$ and satisfies,

$$\frac{\theta(x)}{\log(x)} \to \infty \quad \text{and} \quad \frac{\theta(x)}{x} \to 0, \quad \text{as } x \to \infty. \tag{20}$$

The rule of thumb is that θ needs to be at least of order $\mathcal{O}(\log(n)) \approx \mathcal{O}(\log(1/(1-\rho)))$; this is small compared to the average queue lengths in the system and it becomes negligible as $\rho \uparrow 1$ and $n \uparrow \infty$. Hereafter, we will use θ to denote either $\theta(n)$ or the vector $\theta(n)\mathbf{1}$. The constraint $Q(t) \geq \theta$ implies that $W(t) = CR^{-1}Q(t) \geq CR^{-1}\theta = [\theta \; 2\theta]'$. Effectively, by enforcing safety stock requirements, we are shifting the workload region $\mathcal{S}^*$ to

$$\mathcal{S}_\theta = \{w + CR^{-1}\theta : w \in \mathcal{S}^*\}, \tag{21}$$

and will choose our control to keep $W(t) \in \mathcal{S}_\theta$, $Q(t) \geq \theta$, and also track the target $\Delta(W(t))$.

Tracking policy – definition: For all $t \geq 0$, we observe $Q(t)$ and set $W(t) = CR^{-1}Q(t)$. Reflection: If $W(t) \not\geq CR^{-1}\theta$, we choose a target workload vector that lies in $\mathcal{S}_\theta$, using the controls specified in (13). In particular, we set our target workload for each server i to $W_i^{\theta}(t) = W_i(t) \vee (CR^{-1}\theta)_i$. This is a displacement along the vertical direction of control of U_i^* described in (13) that corresponds to planned idleness of $[(CR^{-1}\theta)_i - W_i(t)]^+$ time units.

Target setting: Next, we pick a target configuration, denoted by $x^\theta(t)$, to hold the workload $W^{\theta}(t) = CR^{-1}\theta + [W(t) - CR^{-1}\theta]^+$ as follows: (a) $CR^{-1}\theta$ assigns θ jobs of each class in the system – these are the safety stocks; and (b) $[W(t) - CR^{-1}\theta]^+$ should be held in its minimum cost configuration given by $\Delta([W(t) - CR^{-1}\theta]^+)$.

It is natural to set the target to $x^\theta(t) = \theta + \Delta([W(t) - CR^{-1}\theta]^+)$. However, this will not work because of the different time scales involved in the control actions of steps (a) and (b). Specifically, whenever $Q(t) \not\geq \theta$, step (a) will try to bring $Q(t)$ back above θ, and this should take roughly $\mathcal{O}(\theta)$ time, whereas, step (b) is reallocating the remaining workload in the system, which will take $\mathcal{O}(n)$ time. Setting the target to $x^\theta(t)$ and choosing allocations to reach it in minimum time, implies that the safety level will also be achieved in $\mathcal{O}(n)$ time; this will not provide an adequate positive drift towards θ in order to maintain this safety level and prevent undesirable system behavior whenever some of the buffers are depleted.

To select a target consistent with (a)–(b), let $l = c\theta$, for some constant $c > 0$, and set

$$x^\theta(t) = \theta + \Psi\left(l; [Q(t) - \theta]^+, \Delta([W(t) - CR^{-1}\theta]^+)\right), \tag{22}$$

where $\Psi\left(l; [Q(t) - \theta]^+, \Delta([W(t) - CR^{-1}\theta]^+)\right)$ is the state l time units into the future along the minimum time trajectory from $[Q(t) - \theta]^+$ to $\Delta([W(t) - CR^{-1}\theta]^+)$.

Tracking: Given the target $x^\theta(t)$, our tracking policy continuously reallocates fractional utilizations in order to reach $x^\theta(t)$ in minimum time using

$$\dot{T}_k(Q(t)) = \beta_k\left(Q(t), x^\theta(t)\right) \cdot \mathbf{1}_{\{Q_k(t)>0\}}, \quad \text{for } k = 1, 2, 3, \tag{23}$$

where $\mathbf{1}_{\{\cdot\}}$ is the indicator function. The structure of these policies is summarize below:

$$Q(t) \longrightarrow W(t) \longrightarrow W^{\theta}(t) \in \mathcal{S}^\theta \longrightarrow x^\theta(t) \longrightarrow \beta\left(Q(t), x^\theta(t)\right). \tag{24}$$

We conclude this section with some general comments. We have proposed a class of "receding horizon tracking" policies. At any given time a target is selected l ($\approx \mathcal{O}(\theta)$) time units into the future, and a control is chosen to reach that target in minimum time. The minimum time path is only used for simplicity; another natural candidate would be to use the path that minimizes total cost in the fluid model. The use of safety stocks is crucial in order to be able to asymptotically achieve the desired fluid trajectories. We make the following remarks on (24): (i) step $W(t) \rightarrow W^{\theta}(t)$ prescribes idleness according to (13); (ii) the mappings $W^{\theta} \rightarrow x^\theta$ and Δ do not prescribe any idleness since $CR^{-1}\Delta(W(t)) = W(t)$; (iii) hence, the system only idles if $W(t) \notin \mathcal{S}^\theta$; (iv) finally, Δ, x^θ, $\beta(Q(t), x^\theta(t))$ and $\dot{T}(Q(t))$

are continuous in $Q(t)$. Asymptotically as $\rho \uparrow 1$ and $n \uparrow \infty$, two things happen: first, $\theta(n)/n \to 0$ and as a result, $\mathcal{S}^{\theta/n} \approx \mathcal{S}^*$ and $W^{\theta/n} \approx W$, and second, $\theta(n) \to \infty$ sufficiently fast such that $\mathbf{P}(Q^n(t) \not\gtrsim \theta(n)) \to 0$ exponentially in θ and thus, polynomially in n. These features imply that the effect of step (a) above is negligible and x^θ will be chosen according to Δ. As a result, we would expect to achieve the desired fluid model behavior and subsequently, the optimal diffusion limit.

4 Heavy traffic asymptotic optimality of CR tracking policy

The main result of the paper is to prove that the proposed tracking policy is asymptotically optimal in the heavy traffic limit. Recall that a policy is described by its allocation process $\{T(t), t \geq 0\}$. Consider a sequence $\{\rho_n\}$ such that $\rho_n \uparrow 1$, and set $n = 1/(1-\rho_n)$. We initialize the network by $Q^n(0) = z_n$ where $|z_n| = n$, and denote by $T^{n,*}$ our tracking policy for the n^{th} system along this sequence.

Theorem 4.1 *Suppose that* $\{T^n\}$ *is any sequence of control policies (one for each* n*). Then,*

$$\liminf_{n\to\infty} \mathbf{E}\left(\int_0^\infty e^{-\gamma t} \sum_k Z^n(t)dt\right) \geq \lim_{n\to\infty} \mathbf{E}\left(\int_0^\infty e^{-\gamma t} \sum_k Z^{n,*}(t)dt\right) = J^* < \infty. \tag{25}$$

A brief outline of the proof is given below. (The details can be found in a full version of this paper available from the author.) Our argument relies on the Bramson-Williams framework that essentially reduces the diffusion analysis to verification of fluid scale properties.

Proposition 4.1 *(Large Deviations estimates) For any* $\epsilon > 0$ *and* n *sufficiently large,*

$$\mathbf{P}\left(\inf_{0\leq s\leq n^2 t} Q^{n,*}(s) \not\gtrsim (1-\epsilon)\theta^n\right) \leq \frac{t}{n^2} \text{ and } \mathbf{P}\left(\inf_{0\leq s\leq n^2 t} W^{n,*}(s) \not\gtrsim (1-\epsilon)CR^{-1}\theta^n\right) \leq \frac{t}{n^2}.$$

Sketch of proof: Follows from (1) and (20) using an exit time calculation as in [24, §6.1].

Proposition 4.2 *(Fluid scale analysis) Along any sequence of initial conditions* $\{z_n\}$ *there exists a converging subsequence* $\{z_{n_j}\}$ *such that almost surely* $(q^{n_j,*}(\cdot), y^{n_j,*}(\cdot), u^{n_j,*}(\cdot)) \to (q(\cdot), y(\cdot), u(\cdot))$ *u.o.c., where* $q(\cdot)$ *is the minimum time trajectory*

$$q(t) = z + (\Delta(CR^{-1}z) - z)\frac{t}{t^*(\Delta(CR^{-1}z), z)}, \quad \text{for } t \leq t^*(\Delta(CR^{-1}z), z),$$

and $q(t) = \Delta(CR^{-1}z)$, *for* $t > t^*(\Delta(CR^{-1}z), z)$.

Sketch of proof: Standard fluid limit arguments, Proposition 4.1, and continuity of Δ and β.

Proposition 4.3 *(State space collapse) For all* $t \geq 0$,

$$\sup_{0\leq s\leq t} |Z^{n,*}(\cdot) - \Delta(W^{n,*}(\cdot)| \to 0, \quad \text{in prob.}$$

Sketch of proof: This is a slight variation of Bramson's result [3, Thm.4] that follows from his Assumption 3.2 [3], which, in turn, is a direct consequence of Proposition 4.2 above.

Proposition 4.4 *(Diffusion limit [27])* $(Z^{n,*}, W^{n,*}, U^{n,*}) \Rightarrow (Z^*, W^*, U^*)$.

Sketch of proof: This result follows from Williams' invariance principle [27, Thm. 4.1] by recognizing that by construction our policy never idles if $W^{n,*} \geq CR^{-1}\theta/n$, the continuity of Δ, the state space collapse condition established in Proposition 4.3, and the fact that the reflection matrix, which is just the identity matrix, is *completely-S*.

Theorem 4.1 Sketch of proof: We first establish a uniform integrability condition (as in [17, §7]) to pass to the limit and establish the RHS of (25). The inequality in the LHS follows from the pathwise optimality of the optimal Brownian control policy.

References

[1] F. Avram, D. Bertsimas, and M. Ricard. Fluid models of sequencing problems in open queueing networks; an optimal control approach. In F. Kelly and R. Williams, editors, *Stochastic Networks*, volume 71 of *Proceedings of the IMA*, pages 199–234. Springer-Verlag, New York, 1995.

[2] M. Bramson. Stability of two families of queueing networks and a discussion of fluid limits. *Queueing Systems*, 28:7–31, 1998.

[3] M. Bramson. State space collapse with applications to heavy-traffic limits for multiclass queueing networks. *Queueing Systems*, 30:89–148, 1998.

[4] H. Chen. Fluid approximations and stability of multiclass queueing networks: work-conserving policies. *Ann. Appl. Prob.*, 5:637–655, 1995.

[5] H. Chen and D. Yao. Dynamic scheduling of a multiclass fluid network. *Oper. Res.*, 41(6):1104–1115, Dec. 1993.

[6] J. G. Dai. On positive Harris recurrence of multiclass queueing networks: A unified approach via fluid limit models. *Ann. Appl. Prob.*, 5:49–77, 1995.

[7] J. G. Dai and S. Meyn. Stability and convergence of moments for multiclass queueing networks via fluid limit models. *IEEE Trans. Aut. Control*, 40(11):1889–1904, Nov. 1995.

[8] J. G. Dai and G. Weiss. Stability and instability of fluid models for certain re-entrant lines. *Math. Oper. Res.*, 21:115–134, 1996.

[9] A. Dembo and T. Zajik. Large deviations: from empirical mean and measure to partial sum processes. *Stoc. Proc. and Appl.*, 57:191–224, 1995.

[10] A. Dembo and O. Zeitouni. *Large Deviations Techniques and Applications.* 2nd ed., Applications of Mathematics, Springer Verlag, 1998.

[11] J. M. Harrison. Brownian models of queueing networks with heterogeneous customer populations. In W. Fleming and P. L. Lions, editors, *Stochastic Differential Systems, Stochastic Control Theory and Applications*, volume 10 of *Proceedings of the IMA*, pages 147–186. Springer-Verlag, New York, 1988.

[12] J. M. Harrison. Balanced fluid models of multiclass queueing networks: a heavy traffic conjecture. In F. Kelly and R. Williams, editors, *Stochastic Networks*, volume 71, pages 1–20. Proceedings of the IMA, 1995.

[13] J. M. Harrison. The BIGSTEP approach to flow management in stochastic processing networks. In F. Kelly, S. Zachary, and I. Ziedins, editors, *Stochastic Networks: Theory and Applications*, pages 57–90. Oxford University Press, 1996.

[14] J. M. Harrison and J. A. Van Mieghem. Dynamic control of brownian networks: State space collapse and equivalent workload formulations. *Ann. Appl. Prob.*, 7:747–771, 1996.

[15] J. M. Harrison and L. M. Wein. Scheduling network of queues: Heavy traffic analysis of a simple open network. *Queueing Systems*, 5:265–280, 1989.

[16] J. M. Harrison and L. M. Wein. Scheduling networks of queues: Heavy traffic analysis of a two-station closed network. *Oper. Res.*, 38(6):1052–1064, Dec. 1990.

[17] S. Kumar. Two-server closed networks in heavy traffic: Diffusion limits and symptotic optimality. *Ann. Appl. Prob.*, 1999. To appear.

[18] H. J. Kushner and P. Dupuis. *Numerical methods for stochastic control problems in continuous time.* Springer, New York, 1992.

[19] H. J. Kushner and L. F. Martins. Heavy traffic analysis of a controlled multiclass queueing network via weak convergence methods. *SIAM J. on Control and Optimization*, 34(5):1781–1797, Sept. 1996.

[20] C. Maglaras. Discrete-review policies for scheduling stochastic networks: Trajectory tracking and fluid-scale asymptotic optimality. *Ann. Appl. Prob.*, 1999. Under revision.

[21] C. Maglaras. Dynamic scheduling in multiclass queueing networks: Stability under discrete-review policies. *Queueing Systems*. 31(3):171–206, 1999.

[22] L. F. Martins, S. E. Shreve, and H. M. Soner. Heavy traffic convergence of a controlled multiclass queueing network. *SIAM J. on Control and Optimization*, 34(6):2133–2171, Nov. 1996.

[23] S. P. Meyn. Stability and optimization of queueing networks and their fluid models. In G. G. Yin and Q. Zhang, editors, *Mathematics of Stochastic Manufacturing Systems*, volume 33 of *Lectures in Applied Mathematics*, pages 175–200. American Mathematical Society, 1997.

[24] A. Shwartz and A. Weiss. *Large Deviations for Performance Analysis: Queues, communications and computing.* Chapman & Hall, 1995.

[25] R. J. Williams. On the approximation of queueing networks in heavy traffic. In F. Kelly, S. Zachary, and I. Ziedins, editors, *Stochastic Networks: Theory and Applications*, pages 35–56. Oxford University Press, 1996.

[26] R. J. Williams. Diffusion approximations for open multiclass queueing networks: Sufficient conditions involving for state space collapse. *Queueing Systems*, 30:27–88, 1998.

[27] R. J. Williams. An invariance principle for semimartingale reflecting Brownian motion. *Queueing Systems*, 30:5–25, 1998.

Scheduling Open Queueing Networks with Sufficiently Flexible Resources*

Sunil Kumar
Graduate School of Business
Stanford University
Stanford, CA 94305-5015
skumar@leland.stanford.edu

Abstract

We study asymptotically optimal scheduling of open queueing networks with flexible non-identical servers with overlapping capabilities, or equivalently, discretionary routing. Extending results obtained by Laws as well as Harrison and Lopez to networks with probabilistic feedback routing, we articulate a condition that ensures that the network is in heavy traffic and the corresponding Brownian control problem is one dimensional. In the interest of brevity, we restrict attention to one example of a network satisfying this condition. We solve the Brownian control problem for this example, and use the solution to (a) construct an asymptotic bound on achievable performance under any policy and (b) propose a Discrete Review scheduling policy. Although we do not rigorously establish that this policy is asymptotically optimal, we present fluid model results that support this conjecture and point towards a method of establishing this result.

1 Introduction

The study of open queueing networks with flexible non-identical servers with overlapping capabilities, or equivalently discretionary routing, is motivated by examples of manufacturing operations such as the following. Consider the job shop shown in Figure 1. It consists of two types of jobs, type 1 that requires only a milling operation, and type 2 that requires a drilling operation and, independent of which resource performed the operation, requires an additional grinding operation with probability 0.5. The job shop contains two machines, CNC1 and CNC2 that can each perform all the required operations, albeit at differing rates. The rates at which each of these machines can carry out the three operations are tabulated in Table 1.

The goal is to find dynamic scheduling policies to control such system. This includes finding a discretionary routing policy that assigns jobs awaiting a particular operation to one of the resources capable of performing that operation as well as a dynamic sequencing policy that tells a resource the order in which jobs awaiting service from that resource must be handled. Assuming linear holding costs, one natural objective of such policy

---

[*] This research is supported in part by a Finmeccanica Faculty Scholarship at Stanford University.

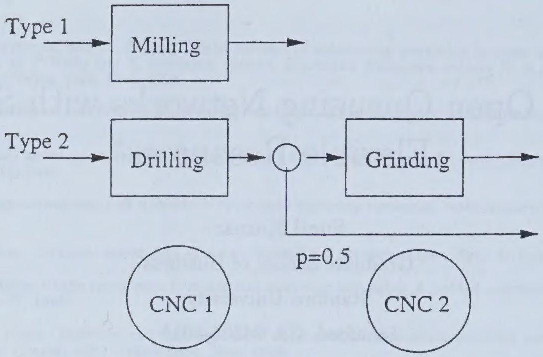


Figure 1: Motivating Example of Job Shop with CNC Machines

Activity	Rate	Activity	Rate
CNC1 Milling	1.150	CNC2 Milling	1.000
CNC1 Drilling	1.100	CNC2 Drilling	1.157
CNC1 Grinding	0.800	CNC2 Grinding	1.300

Table 1: Machine capabilities

design is to find policies that minimize the expected infinite horizon total discounted holding cost of the form

$$J = \mathrm{E}\left(\int_0^\infty e^{-\gamma t} c \cdot Q(t)\, dt\right),$$

where $Q(t)$ is the vector of number of jobs in each class (distinguished by the type of job and the processing step involved) at time t, c is the vector of holding cost rates, and γ is the discount rate.

In general, finding such policies is very hard. It is prudent to relax one's expectation and to find "good" policies that are optimal in some ideal asymptotic regime. A good candidate for such a regime is "heavy traffic" where the system is barely capable of handling the load impressed upon it. A systematic procedure for finding such good policies was proposed by Harrison [3] as follows. First, we set up the system model and the corresponding optimal control problem. Then, we imagine a system parameter n increasing without bound and scale all processes of interest by the parameter n as suggested by functional central limit theorems. We heuristically derive a corresponding Brownian control problem in terms of possible functional limits of scaled processes. We solve the limiting control problem and find an interpretation of the solution of the limiting control problem in the context of the original system. We use this interpretation to come up a with a policy for the original system. Finally, we analyze performance under that policy in the limiting asymptotic regime to verify that it is indeed asymptotically optimal. Of course, this procedure is most useful when the Brownian control problem is easy to solve.

Although the reader may be tempted to believe that designing policies using the above procedure for systems that involve both discretionary routing and sequencing decisions is harder than for those that only involve sequencing decisions, this is not true. Laws [12] and Kelly and Laws [8] showed that in some queueing networks with discretionary routing, namely those that satisfy a condition articulated as the uniqueness of the solution to a linear program, the Brownian control problem is one-dimensional (regardless of the original system dimension) and hence trivial to solve. Rather than using their results for designing asymptotically optimal policies, they chose to concentrate on the qualitative insights provided by their results, in particular on *resource pooling*, in such networks. Harrison and van Mieghem [7] generalized these results on reduction in problem dimension. Harrison and Lopez [6] have used these results to propose a class of policies for systems without feedback routing (parallel server systems). These policies fall in the class of Discrete Review policies proposed by Harrison [4]. However they do not establish asymptotic optimality of these policies. Bell and Williams [1], using the same model of parallel server systems with no feedback routing as [6] but restricting attention to systems with only two servers, propose a continuous review threshold policy that they prove is asymptotically optimal. Recently, Williams [13] has proposed an extension of this threshold policy to parallel server systems without feedback routing but with arbitrarily many servers. On an alternate track, Kushner and Chen [9] find an asymptotically optimal policy for a system with discretionary routing that does not result in a one dimensional limiting control problem.

In this paper, we extend the results of Laws [12] and Harrison and Lopez [6] to networks that involve *probabilistic feedback routing.* That is, the processing sequence may involve a random choice that is not known until the previous step in the sequence is complete. In the example, it is not known whether a grinding step is required until the drilling operation is complete. Also, unlike the models considered in [6, 1, 13], jobs may require more than one processing step (i.e., they may be fed back into the shop) before they exit the system. For such systems with probabilistic feedback routing, we obtain the following results.

- We show that if the obvious analog of the condition articulated by Laws [12] holds, the Brownian control problem is one dimensional.
- When the above condition holds, we derive and solve the limiting control problem. We do this for the case of arbitrary linear holding costs, thus extending the results in Laws [12] along the lines of Harrison and Lopez [6].
- We prove that the solution to the limiting control problem is an asymptotic lower bound on J under any policy. Hence, we show that we only need to construct a policy (or, more generally, a sequence of policies) under which processes of interest converge to the solution to the limiting control problem.
- We propose one such policy called the Discrete Review policy. While we do not provide a proof of convergence, we provide fluid model analysis that suggests the direction of such a proof based on some recent work by Bramson [2] and Williams [14].

In this short paper, we will restrict attention only to the example in Figure 1. The model statement in the next section and all the results in the subsequent sections will be stated with respect to this example. In most cases, the extension to the general case will be obvious, but the reader is advised to refer to the journal paper [11] that is currently under preparation for discussion of the general case.

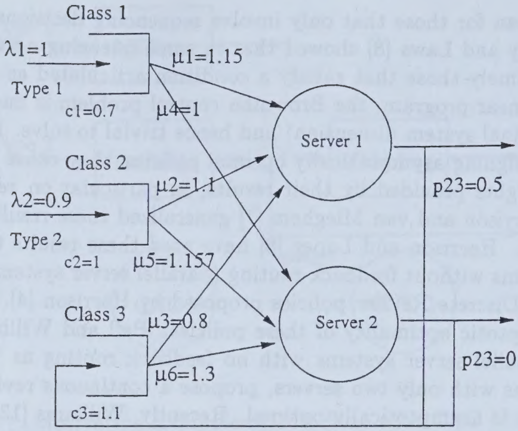


Figure 2: Queueing Network Model of Example

2 Model and Objective

We model the example of the previous section as an open multiclass queueing network with discretionary routing as shown in Figure 2. The network has three classes (with corresponding infinite buffers), two servers, and six "activities" that correspond to a (class, server) pair. That is, activity 1 corresponds to class 1 being processed by server 1 or (1,1). There are external arrivals to classes 1 and 2, corresponding to arrivals of types 1 and 2 in the original system. On completing service, class 1 and class 3 jobs exit the system while class 2 jobs either become class 3 jobs or exit the system with probability 0.5. This occurs regardless of which server actually performed the service.[1] We assume all jobs within a class are served in FIFO fashion. A service time is defined to be the (random) time required to be spent on an activity by the corresponding server in order to complete service of a job in the corresponding class.

Assumption 1 *For simplicity and brevity, we make the following unnecessarily strong assumptions about the primitive stochastic processes of the network.*

- *The arrival processes are independent Poisson processes with rates* $\lambda_1 = 1$ *and* $\lambda_2 = 0.9$ *respectively.*
- *The service times are assumed to be i.i.d. exponential with means given by the reciprocals of the rates tabulated in Table 1. Service times of different activities are independent and independent of arrivals.*
- *Class 2 customers who complete service next become class 3 with probability* $p_{23} = 0.5$ *independent of everything else.*

We assume that the holding costs are class dependent and linear. That is, it costs c_k dollars to hold a job in class k for one time unit. In the example we use,

$$c = [c_1, c_2, c_3]' = [0.9, 1.0, 1.1]'.$$

---

[1] For concreteness, we assume that once a server begins processing of a job, the job stays with that server until its service is complete.

For the rest of the discussion in the paper it is necessary to define two matrices, an input-output matrix R with R_{ki} interpreted as the amount by which class k is reduced on average by devoting one time unit to activity i, and a constituency matrix A with $A_{ri} = 1$ iff activity i is performed by server r. In the example we have

$$R = \begin{bmatrix} \mu_1 & 0 & 0 & \mu_4 & 0 & 0 \\ 0 & \mu_2 & 0 & 0 & \mu_5 & 0 \\ 0 & -p_{23}\mu_2 & \mu_3 & 0 & -p_{23}\mu_5 & \mu_6 \end{bmatrix}, \text{ and}$$

$$A = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 \end{bmatrix}.$$

Scheduling Policy: The notion of an activity allows us to combine both the routing and the sequencing decisions. We specify a scheduling policy by the cumulative time allocation vector

$$T(t) = [T_1(t), T_2(t), ..., T_6(t)]',$$

where $T_i(t)$ is the cumulative time devoted to activity i by the corresponding server in $[0, t]$. For any admissible allocation process T, we need that the cumulative server idleness process[2] $I(t) = et - AT(t)$ be continuous and non-decreasing. Of course, in order to completely specify the class of admissible policies, we need to lay out conditions such as non-anticipation, preemption, etc. But we will refrain from doing so, because we do not need that level of detail in this paper.

Asymptotic Optimality: As stated before, if $Q(t) = [Q_1(t), Q_2(t), Q_3(t)]'$ denotes the vector of contents of each class under an allocation process T, the ideal objective would to find the T that minimizes $J(T) = \mathbf{E}\left(\int_0^\infty e^{-\gamma t} c \cdot Q(t)\, dt\right)$. As argued before, we need to relax our objective, and we settle for the following relaxation. Consider a sequence of systems indexed by n, operating with a sequence of allocation processes $\{T_n\}$. For each n define

$$\widehat{J}_n(T_n) := \mathbf{E}\left(\int_0^\infty e^{-\gamma t} c \cdot \frac{Q(n^2 t)}{n}\, dt\right). \tag{1}$$

Definition 1 *A sequence of admissible allocations $\{T_n^*\}$ (or the policy that results in these allocations) is asymptotically optimal if*

$$\liminf_{n\to\infty} \widehat{J}_n(T_n) \geq \limsup_{n\to\infty} \widehat{J}_n(T_n^*), \tag{2}$$

for any sequence of admissible allocations $\{T_n\}$.

3 Static Allocation Problem

Following Laws [12] and Harrison and Lopez [6], we first solve a static allocation problem that serves two purposes. First, it helps define a notion of heavy traffic under discretionary routing. Second, it allows us to define a Brownian control problem that is one dimensional. The static allocation problem involves solving two LP's that are duals of each other. The primal LP is the following. Choose a 6-vector x and a scalar ρ so as to

$$\begin{aligned} &\text{minimize} && \rho && (3)\\ &\text{subject to} && Rx = \lambda && (4)\\ & && Ax \leq \rho e, \text{ and} && (5)\\ & && x, \rho \geq 0. && (6) \end{aligned}$$

---

[2] In this paper, e always denotes a vector of all ones of appropriate dimension.

The dual LP is the following. Choose a 3-vector y and a 2-vector z so as to

$$\begin{aligned} \text{maximize} \quad & y\lambda & (7)\\ \text{subject to} \quad & yR \leq zA & (8)\\ & ze \leq 1, \text{ and} & (9)\\ & z \geq 0. & (10) \end{aligned}$$

In the example, both LP's have unique solutions. The unique solution to the primal LP (3-6) is $\rho^* = 1$ and $x^* = [0.87, 0.13, 0, 0, 0.654, 0.346]'$. Note that $Ax^* = e$. The interpretation of this result is that the system is barely able to handle the load impressed upon it, and hence can be considered to be in heavy traffic. Also note that $x_3^* = x_4^* = 0$. In the notation of Harrison and Lopez [6], these correspond to non-basic activities. The unique solution to the dual LP (7-10) is $y^* = [0.4238, 0.6402, 0.3943]$ and $z^* = [0.487, 0.513]$. Thus we can see that the static allocation problem for the example satisfies the following assumption that is needed for the results in the rest of the paper.

Assumption 2 *Both the primal LP (3-6) and the dual LP(7-10) have unique solutions. Moreover the solution to primal LP (3-6) has $\rho^* = 1$ and $Ax^* = e$.*

Whenever this assumption is satsfied, the Brownian control problem. This is exploited in the next section.

4 Brownian Control Problem

Using the recipe laid out in [3], and mimicking Harrison and Lopez [6] or Williams [13], one can derive the Brownian control problem corresponding to the desired asymptotically optimal scheduling problem. It is to find a 6-dimensional process of deviation controls $Y(t)$ adapted to a 3-dimensional driftless Brownian motion $X(t)$ so as to

$$\text{minimize } \mathbf{E}\left(\int_0^\infty e^{-\gamma t} c \cdot Z(t)\, dt\right) \tag{11}$$

such that

$$\begin{aligned} & Z(t) = X(t) + RY(t) & (12)\\ & U(t) = KY(t) & (13)\\ & Z_i(t) \geq 0 \text{ for } k = 1,2,3, \text{ and} & (14)\\ & U_k \text{ is non-decreasing with } U_k(0) = 0, \text{ for } k = 1,2,...4. & (15) \end{aligned}$$

where

$$K = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 \end{bmatrix}.$$

Arguing as in Theorem 5.2 of Laws [12], Section 5 of Harrison and Lopez [6], and Section 5 of Williams [13], we have an equivalent control problem in terms of the 1-dimensional process $W(t) = y^* \cdot Z(t)$, where y^* is the unique solution to the dual LP (7-10).

Let $c^* := \min_k \frac{c_k}{y_k^*}$. Then, we have the following lower bound on the instantaneous cost in the Brownian control problem

$$c \cdot Z(t) \geq c^* W(t).$$

If we define the 1-dimensional Brownian motion $\xi(t) = y^* \cdot X(t)$, we have $W(t) = \xi(t) + L(t) \geq 0$ where $L(t)$ is an non-decreasing process with $L(0) = 0$. From the optimality property of the one-sided reflection map in one dimension,

$$W(t) \geq W^*(t) := \xi(t) - \inf_{0 \leq s \leq t} \xi(s).$$

So we must have

$$c \cdot Z(t) \geq c^* W(t) \geq c^* W^*(t).$$

Thus, if we can achieve the instantaneous cost $c \cdot Z(t) = c^* W^*(t)$ in the Brownian control problem (11-15), then we have solved the problem. In the example, $c^* = \frac{c_2}{y_2^*}$. In the example (and in all networks satisfying Assumption 2), the lower bound on instantaneous cost above can indeed be achieved as follows[3]. Set $Z_1^*(t) = Z_3^*(t) = 0$ for all t. This is to be interpreted as saying that good policies never leave too many jobs in classes that have larger values of $\frac{c_k}{y_k^*}$. Set $Z_2^*(t) = \frac{W^*(t)}{y_2^*}$, i.e., hold most jobs in class 2. $U_1^*(t) = 0$ for all t. This is to be interpreted as saying that good policies avoid idling server 1. $U_2^*(t) = -\frac{\inf_{0<s<t} \xi(s)}{z_2^*}$. This implies that U_2^* increases only when $W^*(t) = 0$. That is, server 2 is best idled only when the entire system is close to empty. $U_3^*(t) = U_4^*(t) = 0$ for all t. This is to be interpreted as saying that good policies don't use the non-basic activities 3 or 4 too often.

In a later section, we will strive to construct a policy whose limiting behavior mimics the solution to the Brownian control problem outlined above. But first, we show that that optimal cost function in the Brownian control is an asymptotic lower bound on the cost under any admissible policy.

5 An Asymptotic Performance Bound

Define $J^* := \int_0^\infty e^{-\gamma t} c^* W^*(t)$. We have the following result and its immediate corollary.

Theorem 1 *Under any sequence of allocations, $\{T_n\}$,*

$$\liminf_{n \to \infty} \widehat{J}_n(T_n) \geq J^*. \tag{16}$$

Corollary 1 *If under a sequence of allocations $\{T_n\}$, we have $\lim_{n\to\infty} \widehat{J}_n(T_n) = J^*$, then that sequence of allocations (or the policy that gives rise to these allocations) is asymptotically optimal as defined in Definition 1.*

Outline of Proof of Theorem 1: The main idea is to show that the instantaneous cost process in the solution of the Brownian control problem, $c^* W^*$ is the limit of pathwise lower bounds on the instantaneous cost processes under any policy. A very similar argument is used in the proof of Theorem 7.2 in Kumar [10]. The reader is referred to that paper for the technical details. In order to implement this idea, we observe that we can write $y^* \cdot \frac{Q(n^2t)}{n}$ for any n as $\xi^n(t) + L^n(t) \geq 0$, where L^n is a nondecreasing process with $L^n(0) = 0$. Thus, we must have

---

[3] This optimal solution is not unique.

$$\xi^n(t) + L^n(t) \geq \xi^n(t) - \inf_{0 \leq s \leq t} \xi^n(s), \text{ for each } t. \tag{17}$$

It can be shown that if the sequence of processes ξ^n fail to converge weakly in the Skorohod topology to the Brownian motion ξ, then $\liminf \hat{J}_n(T_n) = \infty$. For sequences of processes that do converge to ξ, the right hand side in (17) converges to W^*, thus yielding (16) since $c \cdot \frac{Q(n^2t)}{n} \geq c^* y^* \cdot \frac{Q(n^2t)}{n}$.

6 Achieving the Desired Diffusion Limit

From Corollary 1, we know that in order to construct an asymptotically optimal sequence of allocations, we only need to construct a sequence of allocations whose costs converge to the optimal cost in the Brownian control problem. In order to do this, following Harrison [4] and Harrison and Lopez [6], we propose a Discrete Review policy for achieving the desired limit. We begin by fixing a review period length ℓ and threshold $\underline{z}_k \geq \ell \max_i \mu_i$ for each class k. In the n-th system, these are of the order of $\log n$. Then, at the beginning of each period, we do the following.

- We review the system state q. We develop a processing plan x, of how to spend server effort in the next ℓ time units, by finding the vector x that

$$\text{minimizes } c \cdot z \text{ subject to} \tag{18}$$

$$\begin{aligned} z &= q + \lambda \ell - Rx & (19)\\ z &\geq \underline{z} & (20)\\ Ax &\leq \ell e & (21)\\ x &\geq 0 & (22) \end{aligned}$$

- We implement this plan by processing $\lfloor (x_k \mu_k) \rfloor$ jobs (in any order) from each class k only from the stock on hand at the beginning of the period. This plan is feasible if $q \geq \underline{z}$[4].
- When the plan is completely implemented, we review state q and repeat for next period.

The reader may be surprised that we did not explicitly use the output of the static allocation LP (3-6) in the planning LP (18-22) of the Discrete Review policy. That is, we are not explicitly enforcing any desired average behavior. We do not have to because this is implicitly contained in the constraints of the planning LP.

7 Fluid Model Analysis of Discrete Review Policy

The reader has no reason to believe that a simple, myopic policy that repeatedly minimizes the instantaneous holding cost of the state at the end of a fixed planning horizon

---

[4] We cannot go into the details here of how the plan is to be modified if it is infeasible.

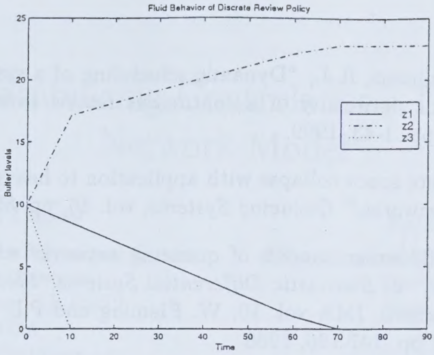


Figure 3: Fluid Model Behavior of Discrete Review Policy

will ever achieve an asymptotic cost of J^*. In light of Corollary 1, and some recent results by Bramson [2] and Williams [14], one could a get clue about the asymptotic optimality of this policy by studying the behavior of this policy in a fluid model, i.e., a model with no variability and continuous dynamics. Figure 3 shows plots of the contents of the three classes as functions of time under the Discrete Review policy for the example considered in this paper in a fluid model.

The following points about the fluid model behavior of this policy are worth noting.

- State Space Collapse: The system achieves a configuration that has fluid only in the right class (class 2) after finite time. Recall that the solution to the Brownian control problem prescribed $Z_1^*(t) = Z_3^*(t) = 0$ for all t. Once this configuration is achieved, it is never left.
- Neither server *ever* idles. In the equilibrium configuration, the processing plan being implemented corresponds to the optimal solution of the static allocation LP (3-6).

In light of these two points, and the work of Bramson [2] and Williams [14], we have both a reason to believe, as well as a direction for establishing a proof of, the asymptotic optimality of the Discrete Review policy as outlined below.

8 Future Work: Proving Asymptotic Optimality

One possible proof of asymptotic optimality suggested by the previous computational example is the following sequence. (1) Show state space collapse in the fluid model. That is, show that in finite time from any initial condition, all the fluid is only contained in the class that has the smallest ratio c_k/y_k^*, and that the servers are never idled in getting to that configuration. (2) Adapt Bramson's work [2] to the class of discrete review policies to show that state space collapse in the fluid model implies state space collapse under Brownian scaling. (3) Adapt Williams's work [14] to the class of discrete review policies to show that state space collapse implies convergence under Brownian scaling. Asymptotic optimality then follows from Corollary 1. This is the subject of [11].

References

[1] Bell, S.L., and Williams, R.J., "Dynamic scheduling of a system with two parallel servers: asymptotic optimality of a continuous review threshold policy in heavy traffic," *Preprint*, pp. 1-43, 1999.

[2] Bramson, M., "State space collapse with application to heavy traffic limits for multiclass queueing networks," *Queueing Systems*, vol. 30, pp. 89-148, 1998.

[3] Harrison, J.M., "Brownian models of queueing networks with heterogeneous customer populations," in *Stochastic Differential Systems: Stochastic Control Theory and Their Applications*, IMA vol. 10, W. Fleming and P.L. Lions (eds.), Springer Verlag, New York, pp. 147-186, 1988.

[4] Harrison, J.M., "The BIGSTEP approach to flow management in stochastic processing networks," in *Stochastic Networks: Theory and Applications*, F.P. Kelly and S. Zachary, and I. Ziedins (eds.), Oxford University Press, pp. 57-90, 1996.

[5] Harrison, J.M., "Heavy traffic analysis of a system with parallel servers: asymptotic optimality of discrete review policies," *Annals of Applied Probability*, vol. 8, pp. 822-848, 1998.

[6] Harrison, J.M., and Lopez, M.J., "Heavy traffic resource pooling in parallel-server systems," to appear in *Queueing Systems*, 1999.

[7] Harrison, J.M., and van Mieghem, J.A., "Dynamic control of Brownian networks: state space collapse and equivalent workload formulations," *Annals of Applied Probability*, vol. 7, pp. 747-771, 1997.

[8] Kelly, F.P., and Laws, C.N., "Dynamic routing in open queueing networks: Brownian models, cut constraints and resource pooling," *Queueing Systems*, vol. 13, pp. 47-86, 1993.

[9] Kushner, H.J., and Chen, Y.N., "Optimal control of assignment of jobs to processors under heavy traffic," *Preprint*, 1998.

[10] Kumar, S., "Two-server closed networks in heavy traffic: diffusion limits and asymptotic optimality," submitted to *Annals of Applied Probability*, 1999.

[11] Kumar, S., "Asymptotically optimal scheduling of open queueing networks with flexible servers using discrete review policies," *In Preparation*, 1999.

[12] Laws, C.N., "Resource pooling in queueing networks with dynamic routing," *Advances in Applied Probability*, vol. 24, pp. 699-726, 1992.

[13] Williams, R.J., "On dynamic scheduling of a call center with complete resource pooling in heavy traffic," to appear in *Fields Institute Communications*, 1999.

[14] Williams, R.J., "Diffusion approximations for open multiclass queueing networks: sufficient conditions involving state space collapse," *Queueing Systems,* vol. 30, pp. 27-88, 1998.

Optimally Stabilizing Controls for a Deterministic Network Model*

Paul Dupuis
Lefschetz Center for Dynamical Systems
Brown University
Providence, R.I. 02912
USA
dupuis@cfm.brown.edu

Rami Atar
Department of Electrical Engineering
The Technion
Haifa, 32000
Israel
atar@ee.technion.ac.il

Abstract

We consider the optimal control of a law of large numbers approximation of a stochastic network. The cost to be minimized is the time till the origin is reached. An explicit formula is given for the minimal cost function, and some of its basic properties are studied.

1 Introduction

This paper considers a problem of service control for a network of servers. Consider such a network, and assume that at each station there are a finite number of distinct customer classes, each with its own buffer. Assume that arrivals to the network and service times are independent of the state of the network. Then under broad conditions one can show that law of large number (LLN) limits exist when time is scaled by a factor n, the state is scaled by $1/n$, and $n \to \infty$. If for purposes of design one uses the limit model in lieu of the original model, then one can often obtain controls that are nearly optimal for the original model in an appropriate sense. Related LLN models have been used previously by a number of authors, incuding [3, 9, 10, 12].

We consider the following problem. Suppose the LLN model is started at a state $x \neq 0$. Then the problem is to find the control (service policy) which drives the state of the model to the origin in minimum time. Such a control will dissipate a large backlog in the original model as quickly as possible. This problem has been solved for the case of re-entrant lines in [12]. For this particular case, however, the minimal time function (the value function for the control problem) is simply a linear function. The class of problems we consider is much more general, and as a consequence the value functions, the corresponding optimizing control laws, and optimally controlled trajectory can be much more interesting. It also turns out that routing controls can also be considered, as long as they appear prior to entry into the network (i.e., not between servers).

Owing to space constraints we focus exclusively in the present paper on the control problem for the limit model. A more complete exposition, including a detailed discussion

---

*This research was supported in part by the National Science Foundation (NSF-DMS-9704426) and the Army Research Office (DAAD19-99-1-0223).

on the form of the limit model that is appropriate for a given original model, will appear in [1]. The main result of the paper is Theorem 3.4, which gives a simple and explicit formula for the value function for the minimum time control problem.

2 Formulation of the Control Problem

In this section we formulate the control problem of interest. As discussed in the introduction, the model we use can be viewed as a law of large numbers approximation to a more detailed stochastic model. We will make use of this connection for interpretive purposes throughout this section. The state space of the process is $\mathbb{R}_+^d$, and one can interpret each of the components as a queue length associated with a specific customer class. The formulation of the model involves two collections of $d-$dimensional vectors. The first are the *directions of constraint*, which we designate by $\{\gamma_i, i = 1, ..., d\}$. These vectors are used to define the Skorokhod or reflection map, which properly corrects the dynamics of the model when one or more components of the state are zero (i.e., one or more customer classes are empty). The second collection will be designated $\{v_j, j = 1, ..., J\}$, and each v_j will characterize the (law of large numbers) evolution of the network when the state of the network is away from $\partial\mathbb{R}_+^d$, and when what we refer to as a "pure" control is applied. For example, consider a model with several stations and several customer classes at each station, and suppose the system can be controlled through the selection of which class should be served at each station. In this case, each v_j will correspond to a service policy that gives all service effort to a single class at each station. Policies that assign a fractional share to different classes can be obtained by taking convex combinations of the v_j.

We will assume the following condition on the directions of constraint. The condition is by now classical in the study of approximations to queueing networks, and is called the Harrison-Reiman condition in [7]. It was first used in [8]. Although the Harrison-Reiman condition is usually associated with single class networks, it also defines the proper Skorokhod Problem for certain formulations of multiclass networks as well. Note that the condition is the original Harrison-Reiman condition, and not the generalization that is also studied in [7].

Condition 2.1 *For each $i \in \{1, ..., d\}$*

$$(\gamma_i)_i = 1, \text{ and } (\gamma_i)_j \leq 0 \text{ for } j \neq i.$$

Let M be the matrix whose ith column is γ_i. Then the spectral radius of M is less than 1.

Let

$$C_{\mathbb{R}_+^d}([0,\infty) : \mathbb{R}^d) \doteq \{\psi \in C([0,\infty) : \mathbb{R}^d) : \psi(0) \in \mathbb{R}_+^d\},$$

where $C([0,\infty) : \mathbb{R}^d)$ is the usual space of continuous functions with the sup norm metric. Let a set of vectors that satisfy Condition 2.1 be given. For each point x on the boundary of $\mathbb{R}_+^d$ let

$$d(x) \doteq \left\{ \sum_{i\in I(x)} a_i\gamma_i : a_i \geq 0, \left\| \sum_{i\in I(x)} a_i\gamma_i \right\| = 1 \right\},$$

where $I(x) \doteq \{i : x_i = 0\}$. The Skorokhod Problem assigns to every path $\psi \in C_{\mathbb{R}_+^d}([0,\infty) : \mathbb{R}^d)$ a path ϕ that starts at $\phi(0) = \psi(0)$, but is constrained to $\mathbb{R}_+^d$ as follows. If ϕ is in the interior of $\mathbb{R}_+^d$ then the evolution of ϕ mimics that of ψ, in that the increments of the two functions are the same until ϕ hits the boundary of $\mathbb{R}_+^d$. When ϕ is on the boundary a constraining "force" is applied to keep ϕ in the domain, and this force can only be applied in one of the directions $d(\phi(t))$, and only for t such that $\phi(t)$ is on the boundary. The precise definition is as follows. For $\eta \in C([0,\infty) : \mathbb{R}^d)$ and $t \in [0,1]$ we let $|\eta|(t)$ denote the total variation of η on $[0,t]$ with respect to the Euclidean norm on $\mathbb{R}^d$.

Definition 2.1 *Let $\psi \in C_{\mathbb{R}_+^d}([0,\infty) : \mathbb{R}^d)$ be given. Then (ϕ,η) solves the SP for ψ (with respect to $\mathbb{R}_+^d$ and $\gamma_i, i = 1,...,d$) if $\phi(0) = \psi(0)$, and if for all $t \in [0,\infty)$*

1. *$\phi(t) = \psi(t) + \eta(t)$,*
2. *$\phi(t) \in G$,*
3. *$|\eta|(t) < \infty$,*
4. *$|\eta|(t) = \int_{[0,t]} 1_{\{\phi(s)\in\partial\mathbb{R}_+^d\}} d|\eta|(s)$,*
5. *There exists a Borel measurable function $\gamma : [0,\infty) \to \mathbb{R}_+^d$ such that $d|\eta|$-almost everywhere $\gamma(t) \in d(\phi(t))$, and such that*
$$\eta(t) = \int_{[0,t]} \gamma(s) d|\eta|(s).$$

Note that η changes only when ϕ is on the boundary, and only in the directions $d(\phi)$.

Under Condition 2.1 the Skorokhod Problem has a solution for all $\psi \in C_{\mathbb{R}_+^d}([0,\infty) : \mathbb{R}^d)$. In addition, the mapping $\psi \to \phi$ (which we call the Skorokhod Map) is Lipschitz continuous [5, 8].

We next define a constrained ordinary differential equation. Under appropriate conditions, this system approximates a stochastic system under a law of large numbers scaling. As is proved in [5], one can define a projection $\pi : \mathbb{R}^d \to \mathbb{R}_+^d$ that is consistent with the constraint directions $\{\gamma_i, i = 1,...,d\}$, in that $\pi(x) = x$ if $x \in \mathbb{R}_+^d$, and if $x \notin \mathbb{R}_+^d$ then $\pi(x) - x = \alpha r$, where $\alpha \geq 0$, $\pi(x) \in \partial\mathbb{R}_+^d$, and $r \in d(\pi(x))$. Figure 1 illustrates the projection for a two dimensional problem.

With this projection given, we can define for each point $x \in \partial\mathbb{R}_+^d$ and each $v \in \mathbb{R}^d$ the *projected velocity* ([4])
$$\pi(x,v) \doteq \lim_{\Delta\downarrow 0} \frac{\pi(x+\Delta v) - \pi(x)}{\Delta}.$$
For details on why this limit is always well defined and further properties of the projected velocity we refer to [4, Section 3 and Lemma 3.8]. Then the ODE of interest is given by
$$\dot{\phi}(t) = \pi(\phi(t), \beta(t)), \tag{1}$$
where
$$\beta(t) = \sum_{j=1}^{J} \rho_j(t) v_j,$$
and for all $t \in [0,\infty)$ $\rho(t) \doteq (\rho_1(t),...,\rho_d(t))$ is a probability vector. The Borel measurable function ρ defines the open loop control that will be applied to the system. To make the

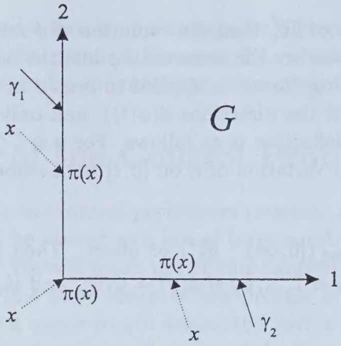


Figure 1: The discrete projection

connection between this control and the actual service assignments (in the special case of service control), we recall that each v_j corresponds to one of the possible pure service assignments. Thus for each $t \in [0, \infty)$ and small $\delta > 0$, $\int_t^{t+\delta} \rho_j(s)ds/\delta$ is (approximately) the fraction of time that the combination of service assignments corresponding to v_j are used on the (scaled) time interval $[t, t+\delta]$. According to the Skorokhod Problem, the velocity β governs the evolution of the network when all states are positive. When one or more states are negative, the projection of the velocity provides the proper correction to the dynamics due to non-negativity constraints.

An absolutely continuous function $\phi : [0, \infty) \to \mathbb{R}_+^d$ is a solution to (1) if the equation is satisfied in an a.e. sense in t. By using the regularity properties of the associated Skorokhod Map, one can prove that all the standard qualitative properties (existence and uniqueness of solutions, stability with respect to perturbations, etc.) hold [5, 6]. In fact, because of the particularly simple nature of the right hand side (i.e., $\pi(\phi(t), \beta(t))$ rather than $\pi(\phi(t), b(\phi(t)) + \beta(t))$ for some function b), one can show that ϕ solves (1) if and only if ϕ is the image of $\psi(t) \doteq \int_0^t \beta(s)ds + x$ under the Skorokhod Map, in which case all such issues become trivial.

The ODE (1) defines the dynamics for the control problem we consider. In order to complete the description, we must also specify a cost structure. The cost we consider is the time for the state to reach the origin, which is to be minimized. Hence for $x \in \mathbb{R}_+^d$ we define

$$V(x) \doteq \inf_{\rho} \tau_x, \tag{2}$$

where $\tau_x \doteq \inf\{s : \phi_x(s) = 0\}$, and ϕ_x is the solution to (1) with initial condition x.

For $V(x)$ to be finite we need some conditions. Define the convex cone

$$\mathcal{C} \doteq \left\{ -\sum_{i=1}^{d} a_i \gamma_i : a_i \geq 0, i = 1, ..., d \right\},$$

which is the negative of the cone of constraint directions that are allowed at the origin. As observed in [4], the cone can be used to characterize stability conditions for (1). The following condition is necessary and sufficient for $V(x) < \infty$ for all $x \in \mathbb{R}_+^d$, and, in the special case of service control, is equivalent to the standard loading condition. Let $\mathcal{C}^\circ$ denote the interior of $\mathcal{C}$.

Condition 2.2 *There exists a probability vector* $\rho = (\rho_1, ..., \rho_J)$ *such that*

$$\sum_{j=1}^{J} \rho_j v_j \in C^{\circ}.$$

Let $D^+V(x)$ (respectively $D^-V(x)$, $DV(x)$) denote the set of superdifferentials (respectively subdifferentials, differential) of V at $x \in \mathbb{R}_+^d$. Given $w \in \mathbb{R}^d$, let $D_wV(x)$ denote the directional derivative in direction w at x:

$$\lim_{\delta \downarrow 0} \frac{V(x + \delta w) - V(x)}{\delta}.$$

We call a function W *radially linear* (on $\mathbb{R}_+^d$) if $W(ax) = aW(x)$ for all $x \in \mathbb{R}_+^d$ and $a \in [0, \infty)$.

3 Representation for the Value Function

The following formula gives an explicit representation for the minimal cost of the control problem stated in the last section. The precise statement is given at the end of this section, and the proof in the section that follows.

$$W(x) = \inf \left\{ \sigma : x + \sigma \sum_{j=1}^{J} \rho_j v_j \in C, \rho_j \in [0, 1], \sum_{j=1}^{J} \rho_j = 1 \right\} \tag{3}$$

Theorem 3.1 *Assume that Condition 2.2 is satisfied and define* $W(x)$ *for* $x \in \mathbb{R}^d$ *by (3). The following conclusions hold.*

1. *W is finite and radially linear on* $\mathbb{R}^d$.
2. *For each* $x \in \mathbb{R}^d$ *the infimum in (3) is achieved at some probability vector* ρ.
3. *W is piecewise linear on* $\mathbb{R}^d$.

Proof: Under Condition 2.2 it is obvious that the cone C can be reached from any starting point x, and so $W(x) < \infty$, while radially linearity is an immediate consequence of the definition of $W(x)$. It follows from the compactness of the unit simplex in $\mathbb{R}^d$ that the infimum is achieved in the definition of $W(x)$. The last property of W stated in Theorem 3.1 is not used later in this paper, and the proof is omitted. ■

Theorem 3.2 *Assume that Condition 2.2 is satisfied and define* $W(x)$ *for* $x \in \mathbb{R}^d$ *by (3). Then* W *is convex on* $\mathbb{R}^d$.

Remark. Since W is convex, directional derivatives exist at all points x and for all directions.

Proof: We begin by considering points x_1 and x_2 such that $W(x_1) = W(x_2) \neq 0$. Let c denote the common value, and let ρ^1 and ρ^2 denote minimizing probability vectors in the expression that defines $W(x_1)$ and $W(x_2)$, respectively. Thus $x_i + c\sum_{j=1}^{J} \rho_j^i v_j \in \partial C$ for $i = 1, 2$. For $\theta \in [0, 1]$, the convexity of C implies

$$\theta x_1 + (1-\theta)x_2 + \theta c \sum_{j=1}^{J} \rho_j^1 v_j + (1-\theta)c \sum_{j=1}^{J} \rho_j^2 v_j = (\theta x_1 + (1-\theta)x_2) + c \sum_{j=1}^{J} (\theta \rho_j^1 + (1-\theta)\rho_j^2) v_j \in C.$$

Since $\theta\rho^1 + (1-\theta)\rho^2$ is a probability vector, it follows that

$$W(\theta x_1 + (1-\theta)x_2) \le c = \theta W(x_1) + (1-\theta)W(x_2).$$

We next consider the case of any points x_1 and x_2 such that $W(x_1) \neq 0$ and $W(x_2) \neq 0$. Let

$$c = \theta W(x_1) + (1-\theta)W(x_2).$$

Since W is radially homogeneous,

$$\begin{aligned} & W(\theta x_1 + (1-\theta)x_2) \\ = \;& W\left(\theta \frac{W(x_1)}{W(x_1)} x_1 + (1-\theta)\frac{W(x_2)}{W(x_2)} x_2\right) \\ = \;& W\left(\left[\frac{\theta W(x_1)}{c}\frac{x_1}{W(x_1)} + \frac{(1-\theta)W(x_2)}{c}\frac{x_2}{W(x_2)}\right] c\right) \\ \le \;& \frac{\theta W(x_1)}{c} W\left(\frac{x_1}{W(x_1)} c\right) + \frac{(1-\theta)W(x_2)}{c} W\left(\frac{x_2}{W(x_2)} c\right) \\ = \;& \theta W(x_1) + (1-\theta)W(x_2). \end{aligned}$$

The case where $W(x_1)$ or $W(x_2)$ equals zero is similar and omitted. ■

Lemma 3.3 *Assume Condition 2.1. Then*

$$C \cap \mathbb{R}^d_+ = \{0\},$$

and therefore $W(x) > 0$ for all $x \in \mathbb{R}^d_+ \backslash \{0\}$.

Proof: The Harrison-Reiman condition implies what is called the *completely-$\mathcal{S}$* condition ([11, 2]), which requires the existence of a vector $v \in \mathbb{R}^d$ satisfying $v_i > 0, i = 1, ..., d$ and $\langle v, \gamma \rangle > 0$ for all $\gamma \in d(0)$. Hence if $y \in C\backslash\{0\}$ then $\langle v, y \rangle < 0$, and so $y \notin \mathbb{R}^d_+$. ■

Theorem 3.4 *Assume that Conditions 2.1 and 2.2 are satisfied and define $W(x)$ for $x \in \mathbb{R}^d$ by (3). Then W equals the value function V defined by (2).*

4 Proof of Theorem 3.4

We first prove some preparatory lemmas. Let

$$A(x) \doteq \left\{ w = \sum_{j=1}^{J} \rho_j v_j : \rho = (\rho_1, ..., \rho_J) \text{ is a probability vector, } x + W(x)w \in C \right\}.$$

Thus $A(x)$ consists of the minimizing velocities in the definition of $W(x)$

Lemma 4.1 *The mapping $x \to A(x)$ is upper semicontinuous.*

Proof: Let $x^i \to x$ as $i \to \infty$, and let $w^i \in A(x^i)$. Assume that $w^i \to w$. Then we must show that $w \in A(x)$. Let ρ^i be probability vectors that satisfy $w^i = \sum_{j=1}^J \rho^i_j v_j$. By taking a subsequence, we can assume $\rho^i \to \rho$. Using the continuity of W,

$$x + W(x) \sum_{j=1}^{J} \rho_j v_j = \lim_{i\to\infty} \left(x_i + W(x_i) \sum_{j=1}^{J} \rho^i_j v_j \right) \in C,$$

and therefore $w = \sum_{j=1}^J \rho_j v_j \in A(x)$. ■

Lemma 4.2 *Assume that Conditions 2.1 and 2.2 are satisfied and define $W(x)$ for $x \in \mathbb{R}^d$ by (3). Consider $x \in \mathbb{R}_+^d \backslash \{0\}$ satisfying $\langle x, e_i \rangle = 0$, and assume that $\langle w, e_i \rangle < 0$ for some $w \in A(x)$. Then for all δ with $|\delta|$ sufficiently small,*

$$W(x + \delta\gamma_i) = W(x).$$

Proof: Let the vectors $\{\gamma_i^*, i = 1, ..., d\}$ be defined by

$$\langle \gamma_i, \gamma_j^* \rangle = \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases}.$$

Condition 2.1 implies the vectors $\{\gamma_i, i = 1, ..., d\}$ are linearly independent, and so this is well defined. The vectors $\{\gamma_i^*, i = 1, ..., d\}$ provide an external representation for C in that

$$C = \left\{ y \in \mathbb{R}^d : \langle y, \gamma_i^* \rangle \leq 0, i = 1, ..., d \right\}.$$

Let $w \in A(x)$ satisfy $\langle w, e_i \rangle < 0$ and let $\sigma = W(x)$. Using the orthogonality relations, for all $k \neq i$

$$\langle x + \sigma w, \gamma_k^* \rangle = \langle x + \delta\gamma_i + \sigma w, \gamma_k^* \rangle.$$

Since $x + \sigma w \in C$, $\langle x + \sigma w, \gamma_k^* \rangle \leq 0$ for all $k = 1, ..., d$. Suppose we can show that $\langle x + \sigma w, \gamma_i^* \rangle < 0$. It will then follow that $\langle x + \delta\gamma_i + \sigma w, \gamma_k^* \rangle \leq 0$ for all $k = 1, ..., d$ and all δ with sufficiently small absolute value. This will imply

$$x + \delta\gamma_i + \sigma w \in C,$$

and consequently $W(x + \delta\gamma_i) \leq W(x)$.

The following expression is easily verified: $\gamma_i^* = -\sum_{k \neq i} \langle e_i, \gamma_k \rangle \gamma_k^* + e_i$. Recall that according to Condition 2.1, $\langle e_i, \gamma_k \rangle \leq 0$ if $i \neq k$. It follows that

$$\begin{aligned} \left\langle x + \sigma w, -\sum_{k \neq i} \langle e_i, \gamma_k \rangle \gamma_k^* + e_i \right\rangle &\leq \langle x + \sigma w, e_i \rangle \\ &= \sigma \langle w, e_i \rangle \\ &< 0, \end{aligned}$$

where we have used the fact $\langle x, e_i \rangle = 0$ and that $\sigma = W(x) > 0$ for $x \in \mathbb{R}_+^d \backslash \{0\}$.

To prove the reverse inequality we use that both the sets C and

$$B(x, \sigma) = \left\{ x + \sigma v : v = \sum_{j=1}^{J} \rho_j v_j : \rho = (\rho_1, ..., \rho_J) \text{ a probability vector} \right\}$$

are convex. It follows from the fact that w is a minimizer that there is a hyperplane of support M to C at $x + \sigma w$ such that

$$[C \backslash M] \cap B(x, \sigma) = \emptyset.$$

Since (as shown above) γ_i^* is not an active constraint, the normal n to M is a linear combination of $\gamma_k^*, k \neq i$, and therefore $\langle n, \gamma_i \rangle = 0$. It follows for any $\delta \in \mathbb{R}$ and $\nu < \sigma$ that

$$C \cap B(x + \delta\gamma_i, \nu) = \emptyset,$$

and therefore $W(x + \delta\gamma_i) \geq \sigma = W(x)$. ■

Lemma 4.3 *Assume that Conditions 2.1 and 2.2 are satisfied and define* $W(x)$ *for* $x \in \mathbb{R}^d$ *by (3). Then the following conclusions hold.*

1. *For* $x \in \mathbb{R}^d_+ \setminus \{0\}$ *and* $w \in A(x)$
$$D_w W(x) = -1.$$

2. *If* $x \in \mathbb{R}^d_+ \setminus \{0\}$, $w \in A(x)$, $\langle x, e_i \rangle = 0$ *and* $\langle w, e_i \rangle < 0$ *then*
$$D_{\gamma_i} W(x) = 0.$$

3. *If* $x \in \mathbb{R}^d$, $w \in \mathbb{R}^d$, $\gamma \in \mathbb{R}^d$, $D_\gamma W(x) = 0$ *and* $a > 0$, *then*
$$D_{w+a\gamma} W(x) \leq D_w W(x).$$

Proof: Consider the first item in the lemma, and let $w \in A(x)$. Then for sufficiently small $\delta > 0$ $W(x) - W(x + \delta w) = \delta$, and it follows that $D_w W(x) = -1$. The second item follows from the preceding lemma, and so we have only to prove the third item.

Let $\delta > 0$ be given. Since W is convex

$$\begin{aligned} W\left(x + \delta\left[\frac{1}{1+a}w + \frac{a}{1+a}\gamma\right]\right) &= W\left(\frac{1}{1+a}[x + \delta w] + \frac{a}{1+a}[x + \delta\gamma]\right) \\ &\leq \frac{1}{1+a}W(x + \delta w) + \frac{a}{1+a}W(x + \delta\gamma) \end{aligned}$$

It follows that

$$D_{\left[\frac{1}{1+a}w + \frac{a}{1+a}\gamma\right]} W(x) \leq \frac{1}{1+a} D_w W(x) + \frac{a}{1+a} D_\gamma W(x)$$

We now use that for any $v \in \mathbb{R}^d$ and $c > 0$

$$D_{cv} W(x) = c D_v W(x)$$

to obtain

$$D_{[w+a\gamma]} W(x) \leq D_w W(x) + D_{a\gamma} W(x) = D_w W(x).$$

■

Proof of Theorem 3.4: We first prove that $W(x) \leq V(x)$. Fix $x \in \mathbb{R}^d_+ \setminus \{0\}$, let $\rho = \rho(t)$ be any control in the definition of $V(x)$, and let $\tau_x > 0$ be the corresponding first time that the origin is reached. Using the definition of the Skorokhod Problem, there exist $a_i(t) \geq 0, i = 1, ..., d, t \in [0, \tau_x]$ such that

$$\dot{\phi}(t) = \sum_{j=1}^{J} \rho_j(t) v_j + \sum_{i=1}^{d} a_i(t) \gamma_i$$

for almost every $t \in [0, \tau_x]$. Integrating over $[0, \tau_x]$ and using the definition $\bar{\rho} \doteq \frac{1}{\tau_x} \int_0^{\tau_x} \rho(t) dt$, we find that

$$-x = \tau_x \sum_{j=1}^{J} \bar{\rho}_j v_j - \nu$$

where $\nu \in \mathcal{C}$, and so $x + \tau_x \sum_{j=1}^{J} \bar{\rho}_j v_j \in \mathcal{C}$. We conclude that $\tau_x \geq W(x)$ and therefore $V(x) \geq W(x)$.

We next prove $W(x) \geq V(x)$. Fix $x \in \mathbb{R}_+^d \setminus \{0\}$. Since Lemma 4.1 shows that $x \to A(x)$ is upper semicontinuous, using the technique of [6, Theorem 3] one can construct a solution to the equation and differential inclusion

$$\begin{aligned} \dot{\phi} &= \pi(\phi, \beta) \\ \beta &\in A(\phi) \end{aligned}$$

and the initial condition $\phi(0) = x$. The equation and inclusion are assumed to hold in an almost everywhere sense. It is automatic that β and ϕ define a control and controlled process that can be used in the definition of $V(x)$. We will show that ϕ reaches the origin by time $W(x)$, which implies that $V(x) \leq W(x)$. To this end, we observe that

$$\dot{\phi}(t) = \beta(t) + \sum_{i=1}^{d} a_i(t)\gamma_i,$$

and that almost surely in t, $a_i(t) > 0$ only when $\langle \phi(t), e_i \rangle = 0$ and $\langle \beta(t), e_i \rangle < 0$. According to Lemma 4.3, this implies

$$D_{\dot{\phi}(t)} W(\phi(t)) = D_{\beta(t) + \sum_{i=1}^{d} a_i(t)\gamma_i} W(\phi(t)) \leq -1$$

for almost every t. Therefore for all times t prior to the first time ϕ hits the origin,

$$W(\phi(t)) - W(x) = \int_0^t D_{\dot{\phi}(s)} W(\phi(s)) ds \leq -t.$$

We conclude that

$$W(\phi(t)) \leq W(x) - t,$$

and therefore ϕ reaches the origin by time $W(x)$. ■

References

[1] R. Atar and P. Dupuis. Explicit solutions for a class of network control problems. Technical report, Brown University, 1999.

[2] A. Bernard and A. El Kharroubi. Régulation de processus dans le premier orthant de $\mathbb{R}^n$. *Stochastics and Stochastics Reports*, 34:149–167, 1991.

[3] D. Bertsimas, I. Paschalidis, and J. Tsitsiklis. Optimization of multiclass queueing networks: polyhedral and nonlinear characterizations of achievable performance. *Annals of Applied Probability*, 4:43–75, 1994.

[4] A. Budhiraja and P. Dupuis. Simple necessary and sufficient conditions for the stability of constrained processes. To appear, *SIAM J. on Applied Math.*

[5] P. Dupuis and H. Ishii. On Lipschitz continuity of the solution mapping to the Skorokhod problem, with applications. *Stochastics*, 35:31–62, 1991.

[6] P. Dupuis and A. Nagurney. Dynamical systems and variational inequalities. *The Annals of Op. Res.*, 44:9–42, 1993.

[7] P. Dupuis and K. Ramanan. Convex duality and the Skorokhod Problem. To appear, *Prob. Th. and Rel. Fields.*

[8] J.M. Harrison and M.I. Reiman. Reflected Brownian motion on an orthant. *The Annals of Probab.*, 9:302–308, 1981.

[9] C. Maglaras. A methodology for dynamic control policy design for stochastic processing networks via fluid models. In *Proceedings of the 36th IEEE Conference on Decision and Control*, New York, 1997. IEEE Publishers.

[10] S. Meyn. Stability and optimization of multiclass queueing networks and their fluid models. Lectures in Applied Mathematics, pages 175–199. American Mathematical Society, Providence, 1997.

[11] M. I. Reiman and R. J. Williams. A boundary property of semimartingale reflecting Brownian motions. *Prob. Theor. Rel. Fields*, 77:87–97, 1988.

[12] G. Weiss. On optimal draining of re-entrant fluid lines. In F. P. Kelley and R. J. Williams, editors, *Stochastic Networks*, volume 71 of *IMA Volumes in Mathematics and Its Applications.* Springer–Verlag, New York, 1995.

Evaluating Cumulant Controllers on a Benchmark Structure Protection Problem in the Presence of Classic Earthquakes*

Khanh D. Pham	Michael K. Sain	Stanley R. Liberty	B. F. Spencer, Jr.
Dept. Electrical Eng.	Dept. Electrical Eng.	Provost and VPAA	Dept. Civil Eng. & Geo Sci
Univ. of Notre Dame	Univ. of Notre Dame	Bradley University	Univ. of Notre Dame
Notre Dame, IN 46556	Notre Dame, IN 46556	Peoria, IL 61625	Notre Dame, IN 46556

Abstract

The topic of Risk Sensitive (RS) control is receiving substantial and continued research from the theoretical community oriented toward stochastic control theory. As yet, however, RS control design has not made a substantial entry into control applications. Moreover, its properties have not yet been compared in any detail with those of other paradigms for control thinking. If one regards the RS optimization criterion as a denumerable linear combination of cumulants, then it is seen as a very interesting case of control design based upon linear combinations of cumulants. In this guise, the problem class goes back to the mid-1960s, when linear combinations of the first two cost cumulants, mean cost and cost variance, were first studied at this Allerton Conference. The present paper reports the application of cost cumulant control, based upon the first two, and the first three, cumulants to control of the Benchmark Structural Control Problem (http://www.nd.edu/~quake) recently studied by numerous authors using a variety of different control algorithms. Although it is definitely the novice in the competition, cost cumulant control performs quite respectably. Moreover, the present results involving the first three cost cumulants appear to be the first of their type to appear in the literature.

1 Introduction

In 1964 Sain first reported research results on optimal stochastic control problems in which the optimization criteria involved both the variance and the mean of the cost functional (performance measure) used in the problem statements [12, 13, 14]. Since then, Sain and colleagues have investigated a variety of issues within this problem class, which has become known as minimum-cost-variance (MCV) control—a class in which one seeks control laws which minimize the cost variance while enforcing a cost mean constraint [8, 9, 10, 11, 7, 15, 16, 18, 21, 19, 20, 22, 23, 24, 25]. The task of minimizing cost variance by itself, without a constraint on the mean cost, may lead to unbounded mean costs. Therefore, it is natural to constrain the mean, which leads directly to the notion of linear combinations of cost cumulants, the context of this paper. A natural extension of this thinking is to the case of denumerable linear combinations of cumulants, when the sum exists; and this leads to the connection with risk-sensitive control.

In Liberty [4], for a linear system with Gaussian-white input noise and Gaussian initial state, all cumulants of any finite-time integral-quadratic-form in the system state

---

[*] This work was supported in part by the Frank M. Freimann Chair in Electrical Engineering, University of Notre Dame, and by the National Science Foundation under Grants CMS 93-01584, CMS 95-00301, CMS 95-28083 and CMS 99-00234.

were shown to be quadratic-affine in the initial state mean. Subsequently Liberty and Hartwig [5] showed that these cumulants could be calculated by solving coupled matrix differential equations and Liberty [6] developed *ad hoc* feedback control laws from the equations associated with the first two cumulants and observed in examples that these feedback control laws performed as one might expect MCV control laws to perform. In the sections which follow, the design considerations evolve within just this type of structure.

For convenience, the paper contains a brief discussion of the k-cost-cumulant control paradigm. There follows a sketch of the Benchmark Structural Control Problem (http://www.nd.edu/~quake), in which a realistic three-degree-of-freedom structure, under excitation by various earthquakes, is protected by an active mass damper (AMD). Application of the cost-cumulant control design equations to the Benchmark Problem is approached by means of a perturbation analysis of the control design equations.

2 A Few Preliminary Remarks

Consider a linear dynamical system modeled on the time interval $T = [t_0,\ t_f]$ by

$$dx(t) = F(t)x(t)dt + G(t)dw(t) \tag{1}$$

where the system state $x(t) \in \mathcal{R}^n$ and the noise $w(t) \in \mathcal{R}^p$ is a stationary Wiener process with correlation of increments $E\{[w(t) - w(\sigma)][w(t) - w(\sigma)]^T\} = \tilde{W}|t - \sigma|$. Assume that the initial state $x(t_0)$ is non-random. Associate with (1) a functional

$$J = x^T(t_f)Q_f x(t_f) + \int_{t_0}^{t_f} [x^T(t)N(t)x(t)]dt \tag{2}$$

where Q_f and $N(t)$ are symmetric and positive semi-definite. Also, assume that $F(\cdot)$, $G(\cdot)$, and $N(\cdot)$ are continuous on T. For the above system and associated functional J, from [5] the equations which determine the k^{th} cumulant of J are

$$\kappa_k = x_0^T H(t_0, k)x_0 + Tr\big\{D(t_0, k)\big\}, \tag{3}$$

where x_0 is the initial system state mean; $Tr\{\cdot\}$ denotes the trace of the enclosed matrix and $H(t_0, i)$ and $D(t_0, i)$ are solutions at $t = t_0$ of the coupled matrix diff equations:

$$\frac{d}{dt}H(t,1) = -F^T(t)H(t,1) - H(t,1)F(t) - N(t) \tag{4}$$

$$\frac{d}{dt}H(t,i) = -F^T(t)H(t,i) - H(t,i)F(t) - \sum_{j=1}^{i-1}\frac{2i!}{j!(i-j)!}H(t,j)G(t)\tilde{W}G^T(t)H(t,i-j) \tag{5}$$

$$\frac{d}{dt}Tr\big\{D(t,i)\big\} = -Tr\big\{H(t,i)G(t)\tilde{W}G^T(t)\big\}, \qquad i \geq 1 \tag{6}$$

with terminal conditions $H(t_f, 1) = Q_f$, $H(t_f, i) = 0$ for $i \geq 2$, and $D(t_f, i) = 0$ for $i \geq 1$.

3 The k-Cost-Cumulant Control Paradigm

Now consider a linear dynamical system with state $x_c(t) \in \mathcal{R}^n$ modeled on T by

$$dx_c(t) = \big(A(t)x_c(t) + B(t)u(t)\big)dt + G(t)dw(t) \tag{7}$$

where $A(\cdot)$, and $B(\cdot)$ are continuous on T and the control, $u(t) \in \mathcal{R}^m$ is a linear, memoryless, full-state feedback control given by

$$u(t) = K(t)x_c(t) \tag{8}$$

where $K(\cdot)$ is any $m \times n$ matrix feedback gain which is in the class $\mathcal{K}_k$ of admissible feedback gains defined in Definition 1 below. The process $w(\cdot)$ and the coefficient $G(\cdot)$ for the system described by (7) are the same as in (1). Also, we assume that both (1) and (7) have the same known initial condition, i.e., $x_c(t_0) = x(t_0)$.

Attach to (7) the cost functional

$$J_c = x_c^T(t_f)Q_f x_c(t_f) + \int_{t_0}^{t_f} [x_c^T(\tau)Q(\tau)x_c(\tau) + u^T(\tau)R(\tau)u(\tau)]d\tau \tag{9}$$

where $Q(\tau)$ and $R(\tau)$ are symmetric and continuous on T with $Q(\tau)$ positive semidefinite and $R(\tau)$ positive definite. Also Q_f is the same as in (2). Letting $F(\cdot)$ and $N(\cdot)$ in (1) and (2) be given by $F(t) = A(t) + B(t)K(t)$ and $N(t) = Q(t) + K^T(t)R(t)K(t)$. So, for every feedback gain $K(\cdot)$ in $\mathcal{K}_k$ then $x(\cdot) = x_c(\cdot)$ and $J = J_c$. Now the class, $\mathcal{K}_k$ of admissible feedback gains can be defined as follows:

Definition 1 *Let $\overline{K}$ be a compact subset of the vector space of all $m \times n$ real matrices. ($\overline{K}$ is the allowable set of gain values.) Let $\mathcal{K}_k$ be the set of all $m \times n$ matrix functions which are continuous on T with values $K(t) \in \overline{K}$, and which yield existence of solutions on T to the k coupled, matrix differential equations (4) through (6) with the terminal conditions specified above.*

Form the partitioned matrix $Y^T(t) = [H^T(t,1)\ D^T(t,1) \dots H^T(t,k)\ D^T(t,k)]$ and let $y(t) = vec[Y(t)]$ where the operation $vec[Y(t)]$ produces the $2kn^2 \times 1$ partitioned matrix with the columns of $Y(t)$ as entries. Also notate the relationship among $\dot{Y}$, Y, and K as described in (4) through (6) with the specified terminal conditions by

$$\frac{d}{dt}Y(t) = f(t, Y(t), K(t)), \qquad Y(t_f) = Y_f. \tag{10}$$

Definition 2 *Let performance index be defined as $\phi_0(t_0, Y(t_0); x(t_0); \mu_1, \dots, \mu_k) = \sum_{i=1}^{k} \mu_i \kappa_i$ where each μ_i is a non-negative real scalar constant.*

Without loss of generality, set $\mu_1 = 1$. The k-cumulant optimization problem can now be seen as the minimization of $\min\limits_{K(\cdot) \in \mathcal{K}_k} \phi_0\big(t_0, Y(t_0); x(t_0); \mu_2, \dots, \mu_k\big)$ subject to (10).

4 k-Cost-Cumulant Control Design Equations

Under the feedback gain specified in (8), the design equations (4-5) become

$$\begin{aligned}\dot{H}(t,1) = {} & -A^T(t)H(t,1) - H(t,1)A(t) + H(t,1)B(t)R^{-1}(t)B^T(t)H(t,1) \\ & -Q(t) - \sum_{i=2}^{k} \mu_i H(t,i)B(t)R^{-1}(t)B^T(t)\sum_{j=2}^{k}\mu_j H(t,j)\end{aligned} \tag{11}$$

$$\begin{aligned}\dot{H}(t,i) = {} & -A^T(t)H(t,i) - H(t,i)A(t) + H(t,1)B(t)R^{-1}(t)B^T(t)H(t,i) \\ & + \sum_{j=2}^{i} \mu_j H(t,j)B(t)R^{-1}(t)B^T(t)H(t,i) + H(t,i)B(t)R^{-1}(t)B^T(t)H(t,1) \\ & + H(t,i)B(t)R^{-1}(t)B^T(t)\sum_{j=2}^{i}\mu_j H(t,j) - \sum_{j=1}^{i-1}\frac{2i!}{j!(i-j)!}H(t,j)G(t)\tilde{W}G^T(t)H(t,i-j), i \geq 2.\end{aligned} \tag{12}$$

From [8, 9, 10], the optimal feedback gain which minimizes performance index follows

$$K(t) = -R^{-1}(t)B^T(t)\big[H(t,1) + \sum_{i=2}^{k} \mu_i H(t,i)\big]. \tag{13}$$

5 Earthquake-Protection Structure Benchmark

The structure described in [17] on which the evaluation model is based is an actively controlled, three-story, single-bay, model building considered in [2, 3]. The test structure, shown in photo form in Figure 1 and in diagram form in Figure 2, is designed to be a scale model of the prototype building discussed in [1] and is subject to one-dimensional ground motion. For control purposes, a simple implementation of an AMD was placed on the third floor of the structure. The AMD consists of a single hydraulic actuator with steel masses attached to the ends of the piston rod. See Figure 3 for a close-up photo.

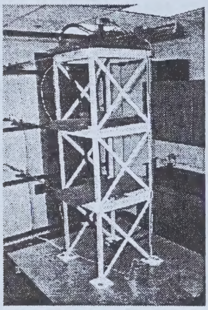


Figure 1: AMD Benchmark Structure.

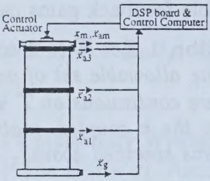


Figure 2: Schematic Diagram of Experiment Setup.

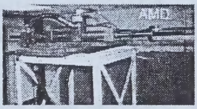


Figure 3: Active Mass Driver.

The cumulant controllers framework is now applied to the test structure with AMD system. The 28-state evaluation model of the test structure is then of the form

$$\dot{x}(t) = Ax(t) + Bu(t) + E\ddot{x}_g(t); \qquad y(t) = C_y x(t) + D_y u(t) + F_y \ddot{x}_g(t) + v \tag{14}$$

$$z(t) = C_z x(t) + D_z u(t) + F_z \ddot{x}_g(t) \tag{15}$$

where x is the state vector, $\ddot{x}_g$ is the earthquake excitation, u is the control action, v is the corrupting measurement white noise $y = [x_m, \ddot{x}_{a1}, \ddot{x}_{a2}, \ddot{x}_{a3}, \ddot{x}_{am}, \ddot{x}_g]^T$ the directly measured responses, and $z = [x_1, x_2, x_3, x_m, \dot{x}_1, \dot{x}_2, \dot{x}_3, \dot{x}m, \ddot{x}_{a1}, \ddot{x}_{a2}, \ddot{x}_{a3}, \ddot{x}_{am}]^T$ is the vector of responses that can be regulated. Here x_i is the displacement of the ith floor relative to the ground, x_m is the displacement of the AMD relative to the third floor, $\ddot{x}_{ai}$ is the absolute acceleration of the ith floor, and $\ddot{x}_{am}$ is the absolute acceleration of the AMD mass. The coefficient matrices in equations (14-15) are determined from the data collected at the Structural Dynamics and Control/Earthquake Engineering Laboratory (SDC/EEL) at the University of Notre Dame.

The first type of excitation will be a stationary random process with a spectral density defined by the Kanai-Tajimi spectrum:

$$S_{\ddot{x}_g\ddot{x}_g} = \frac{S_0(4\zeta_g^2\omega_g^2\omega^2 + \omega_g^4)}{(\omega^2 - \omega_g^2)^2 + 4\zeta_g^2\omega_g^2\omega^2}; \quad S_0 = \frac{0.03\zeta_g}{\pi\omega_g(4\zeta_g^2+1)}\, g^2\, s; \quad \begin{array}{r} 20\, rad/s \le \omega_g \le 120\, rad/s \\ 0.3 \le \zeta_g \le 0.75 \end{array} \tag{16}$$

The first and second evaluation criteria which evaluate a controller's ability to minimize the maximum interstory drift and the floor rms acceleration are given

$$J_1 = \max_{\omega_g,\,\zeta_g}\left\{\frac{\sigma_{d_1}}{\sigma_{x30}}, \frac{\sigma_{d_2}}{\sigma_{x30}}, \frac{\sigma_{d_3}}{\sigma_{x30}}\right\}; \qquad J_2 = \max_{\omega_g,\,\zeta_g}\left\{\frac{\sigma_{\ddot{x}_{a1}}}{\sigma_{\ddot{x}_{a30}}}, \frac{\sigma_{\ddot{x}_{a2}}}{\sigma_{\ddot{x}_{a30}}}, \frac{\sigma_{\ddot{x}_{a3}}}{\sigma_{\ddot{x}_{a30}}}\right\} \tag{17}$$

where σ_{d_i} and $\sigma_{\ddot{x}_{ai}}$ are the rms interstory drift and acceleration of the ith floor, respectively. The interstory drifts are defined as follows $d_1(t) = x_1(t)$, $d_2(t) = x_2(t) - x_1(t)$, and $d_3(t) = x_3(t) - x_2(t)$. Also $\sigma_{x_{30}} = 1.31cm$, $\sigma_{\ddot{x}_{a30}} = 1.79\,g$ are the worst case rms displacement and rms acceleration of the third floor for the uncontrolled building (occuring when $\omega_g = 37.3rad/sec$, $\zeta_g = 0.3$). The third, fourth and fifth evaluation criteria are used to evaluate the control resources used by the candidate controllers.

$$J_3 = \max_{\omega_g,\,\zeta_g}\left\{\frac{\sigma_{x_m}}{\sigma_{x30}}\right\}; \qquad J_4 = \max_{\omega_g,\,\zeta_g}\left\{\frac{\sigma_{\dot{x}_m}}{\sigma_{\dot{x}30}}\right\}; \qquad J_5 = \max_{\omega_g,\,\zeta_g}\left\{\frac{\sigma_{\ddot{x}_{am}}}{\sigma_{\ddot{x}_{a30}}}\right\}. \tag{18}$$

where $\sigma_{\dot{x}_{30}} = 47.9cm$ the worst case rms velocity of the third floor for the uncontrolled building (occuring when $\omega_g = 37.3rad/sec$, $\zeta_g = 0.3$). The hard constraint for the control effort are $\sigma_u \leq 1V$, $\sigma_{\ddot{x}_{am}} \leq 2g's$ and $\sigma_{x_m} \leq 3cm$.

In addition to the Kanai-Tajimi spectrum excitation, the benchmark controllers are also evaluated based on their abilities to minimize the nondimensionalized peak interstory drifts and peak floor accelerations using the historical earthquake records of the 1940 El Centro (Centro) and 1968 Hachinohe (Hach) earthquakes.

$$J_6 = \max_{\substack{\text{Centro}\\ \text{Hach}}}\left\{\max_{\text{t}}\left\{\frac{|d_1|}{x_{3o}}, \frac{|d_2|}{x_{3o}}, \frac{|d_3|}{x_{3o}}\right\}\right\}; \; J_7 = \max_{\substack{\text{Centro}\\ \text{Hach}}}\left\{\max_{\text{t}}\left\{\frac{|\ddot{x}_{a1}|}{\ddot{x}_{a3o}}, \frac{|\ddot{x}_{a2}|}{\ddot{x}_{a3o}}, \frac{|\ddot{x}_{a3}|}{\ddot{x}_{a3o}}\right\}\right\} \tag{19}$$

The control constraints are $\max_{\text{t}} |u(t)| \leq 3V$, $\max_{\text{t}} |x_m(t)| \leq 9cm$, and $\max_{\text{t}} |\ddot{x}_{am}(t)| \leq 6g$. Again the assessment of control resource used by candidate controllers is based on

$$J_8 = \max_{\substack{\text{Centro}\\ \text{Hach}}}\left\{\max_{\text{t}}\left\{\frac{|x_m|}{x_{3o}}\right\}\right\}; \; J_9 = \max_{\substack{\text{Centro}\\ \text{Hach}}}\left\{\max_{\text{t}}\left\{\frac{|\dot{x}_m|}{\dot{x}_{3o}}\right\}\right\}; \; J_{10} = \max_{\substack{\text{Centro}\\ \text{Hach}}}\left\{\max_{\text{t}}\left\{\frac{|\ddot{x}_{am}|}{\ddot{x}_{a3o}}\right\}\right\} \tag{20}$$

For the El Centro earthquake, $x_{3o} = 3.37\,cm$, $\dot{x}_{3o} = 131\,cm/sec$, and $\ddot{x}_{a3o} = 5.05\,g's$. For the Hachinohe earthquake, $x_{3o} = 1.66\,cm$, $\dot{x}_{3o} = 58.3\,cm/sec$, and $\ddot{x}_{a3o} = 2.58\,g's$.

6 Local Design by Perturbation

Having been introduced in a prior section, the cost-cumulant controller design equations equations, of course, are nonlinear, and so represent an interesting study in themselves. In order to assess the trends of performance offered by the used of second and third cumulants, we choose in the present instance a perturbation approach to the application of the equations. At steady state, the 3-cumulant controller has the form

$$K = -R^{-1}B^T\left[H_1 + \mu_2 H_2 + \mu_3 H_3\right] \tag{21}$$

where H_1, H_2, and H_3 are solutions to coupled algebraic Riccati-type matrix equations

$$0 = A^T H_1 + H_1 A + Q - H_1 S H_1 + \mu_2^2 H_2 S H_2 + \mu_2\mu_3 H_2 S H_3 + \mu_2\mu_3 H_3 S H_2 + \mu_3^2 H_3 S H_3 \tag{22}$$

$$0 = A^T H_2 + H_2 A - 2\mu_2 H_2 S H_2 - H_2 S H_1 - H_1 S H_2 - \mu_3 H_2 S H_3 - \mu_3 H_3 S H_2 + 4 H_1 W H_1 \tag{23}$$

$$\begin{aligned} 0 = {} & A^T H_3 + H_3 A - 2\mu_3 H_3 S H_3 - H_3 S H_1 - H_1 S H_3 - \mu_2 H_3 S H_2 - \mu_2 H_2 S H_3 \\ & + 6 H_1 W H_2 + 6 H_2 W H_1 \end{aligned} \tag{24}$$

where $S = BR^{-1}B^T$ and $W = G\tilde{W}G^T$. Here the general second order Taylor series expansion for solutions to equations (22) through (24) is of the form

$$\tilde{H}_i(\mu^* + \mu) = H_i(\mu^*) + \nabla^T H_i(\mu^*)\mu + \frac{1}{2!}\mu^T \nabla^2 H_i(\mu^*)\mu + O^{(3)}(\zeta), \quad i = 1, 2, 3. \tag{25}$$

where the local point of expansion $\mu^* = [0\ 0]^T$, $\mu = [\mu_2\ \mu_3]^T$ and ζ satisfy

$$\zeta = \lambda\mu^* + (1-\lambda)(\mu^* + \mu), \qquad 0 \le \lambda \le 1. \tag{26}$$

The gradient and the Hessian matrices of $H_i(\mu)$ are by definition

$$\nabla H_i(\mu) = \begin{bmatrix} \frac{\partial H_i}{\mu_2} \\ \frac{\partial H_i}{\mu_3} \end{bmatrix}; \qquad \nabla^2 H_i(\mu) = \begin{bmatrix} \frac{\partial^2 H_i}{\partial \mu_2^2} & \frac{\partial^2 H_i}{\partial \mu_3 \partial \mu_2} \\ \frac{\partial^2 H_i}{\partial \mu_2 \partial \mu_3} & \frac{\partial^2 H_i}{\partial \mu_3^2} \end{bmatrix}. \tag{27}$$

We introduce the notations $H_{i0} = H_i(\mu)\ |_{\mu=0}$; $H_{i1} = \frac{\partial H_i(\mu)}{\partial \mu_2}\ |_{\mu=0}$; $H_{i2} = \frac{\partial H_i(\mu)}{\partial \mu_3}\ |_{\mu=0}$; $H_{i3} = \frac{\partial^2 H_i(\mu)}{\partial \mu_2^2}\ |_{\mu=0}$; $H_{i4} = \frac{\partial^2 H_i(\mu)}{\partial \mu_3 \partial \mu_2}\ |_{\mu=0} = \frac{\partial^2 H_i(\mu)}{\partial \mu_2 \partial \mu_3}\ |_{\mu=0}$; $H_{i5} = \frac{\partial^2 H_i(\mu)}{\partial \mu_3^2}\ |_{\mu=0}$.

The approximate steady state solutions to equations (22-24) now become

$$\tilde{H}_1(\mu_2, \mu_3) \cong H_{10} + H_{11}\mu_2 + H_{12}\mu_3 + \frac{1}{2}H_{13}\mu_2^2 + H_{14}\mu_2\mu_3 + \frac{1}{2}H_{15}\mu_3^2 \tag{28}$$

$$\tilde{H}_2(\mu_2, \mu_3) \cong H_{20} + H_{21}\mu_2 + H_{22}\mu_3 + \frac{1}{2}H_{23}\mu_2^2 + H_{24}\mu_2\mu_3 + \frac{1}{2}H_{25}\mu_3^2 \tag{29}$$

$$\tilde{H}_3(\mu_2, \mu_3) \cong H_{30} + H_{31}\mu_2 + H_{32}\mu_3 + \frac{1}{2}H_{33}\mu_2^2 + H_{34}\mu_2\mu_3 + \frac{1}{2}H_{35}\mu_3^2 \tag{30}$$

7 Simulation Results

It is common practice also to report statistics on peak, as well as rms quantities. For peak statistics, different earthquakes of classic repute are employed, in this case El Centro and Hachinohe. These classic earthquakes are chosen for their special dynamic characteristics; and a collection of them can give quite a good dynamic test for a protected building model. For the present study, the design parameters of the sample controller provided with the benchmark were used. Using this control as a baseline, several 2-cumulant and 3-cumulant controllers were designed for different values of μ_2 and μ_3 and evaluated.

In the following tables, presented in parentheses below each result is the percent reduction achieved when compared to the baseline controller. The cumulant controllers produce significant response reductions.

Table 1 shows that the results of controller A are comparable to those achieved by controller B. Moreover, in Table 2 of the 3-cumulant control case, there is significantly greater freedom over the baseline and 2-cumulant control cases in affecting the rms and peak responses. The results from table 2 also illustrate the possibility of process noise covariance as a design parameter. The larger the percent increase of covariance, the better the performance becomes.

8 Conclusions

This paper has applied cost-cumulant control design to the first of the American Society of Civil Engineer's benchmark problems for protection of civil structures from earthquakes.

The results compare well with prior work, available on the website. This appears to be the first substantial instance in which the effect of the third cost cumulant can be assessed. The results indicate in a most encouraging way that the third cumulant may be able to support both the second and the first cumulants in achieving desirable control system behaviors, though of course not with the same level of influence.

References

[1] L. L. Chung, R. C. Lin, T. T. Soong and A. M. Reinhorn, "Experiments on Active Control for MDOF seismic structures," *Journal of Engineering Mechanics,* Vol. 115, No. 8, pp. 1609-1627, 1989.

[2] S. J. Dyke, B. F. Spencer, Jr., A. E. Belknap, K. J. Ferrell, P. Quast and M. K. Sain, "Absolute Acceleration feedback Control Strategies for the Active Mass Driver," *Proceedings 1st World Conference on Structural Control,* Vol. 2, pp. TP1:51-TP1:60, Pasadena, California, 3-5 August 1994.

[3] S. J. Dyke, B. F. Spencer, Jr., P. Quast, D. C. Kaspari, Jr. and M. K. Sain, "Implementation of an Active mass Driver using Acceleration Feedback Control," *Microcomputers in Civil Engineering,* Vol. 11, pp.305-323, 1996.

[4] S. R. Liberty, "Characteristic Functions of LQG Control," *Ph.D. Dissertation,* Department of Electrical Engineering, University of Notre Dame, August 1971.

[5] S. R. Liberty and R. C. Hartwig, "On the Essential Quadratic Nature of LQG Control-Performance Measure Cumulants," *Information and Control,* Vol. 32, No. 3, pp. 276-305, November 1976.

[6] S. R. Liberty, "Performance Analysis of Stochastic Linear Control Systems: A New Viewpoint," *Proceedings International Forum on Alternatives for Linear Multivariable Control,* pp. 79-86, October 1977.

[7] S. R. Liberty, P. Quast, and M. K. Sain, "Control Access to Performance Measure Statistics: A Second Moment Case," *Proceedings Thirty-Fifth Allerton Conference on Communication, Control, and Computing,* pp. 933-942, October 1997.

[8] K. D. Pham, "Design of k^{th}-Order Linear Optimal Cost Cumulant Controller," M. S. Thesis, Department of Electrical Engineering, University of Nebraska, August 1998.

[9] K. D. Pham, S. R. Liberty, and M. K. Sain, "Linear Optimal Cost Cumulant Control: A k-Cumulant Problem Class," *Proceedings Thirty-Sixth Allerton Conference on Communication, Control, and Computing,* pp. 460-469, October 1998.

[10] K. D. Pham, S. R. Liberty, M. K. Sain, and B. F. Spencer, Jr., "Generalized Risk Sensitive Building Control: Protecting Civil Structures with Multiple Cost Cumulants," *Proceedings American Control Conference,* pp. 500-504, June 1999.

[11] P. Quast, "Output Feedback Minimum Cost Variance Control Schemes with Application To Seismic Protection of Civil Engineering Structures," *Ph.D. Dissertation,* Department of Electrical Engineering, University of Notre Dame, Notre Dame, Indiana, September 1997.

[12] M. K. Sain, "On Minimal-Variance Control of Linear Systems with Quadratic Loss," *Ph.D. Dissertation,* Department of Electrical Engineering and Coordinated Science Laboratory, University of Illinois, Urbana, Illinois, January 1965.

[13] M. K. Sain, "Relative Costs of Mean and Variance Control for a Class of Linear, Noisy Systems," *Proceedings Allerton Conference on Circuit and System Theory,* pp. 121-129, 1965.

[14] M. K. Sain, "Control of Linear Systems According to the Minimal Variance Criterion-A New Approach to the Disturbance Problem," *IEEE Transactions on Automatic Control,* AC-11, No. 1, pp. 118-122, January 1966.

[15] M. K. Sain, C.-H. Won, and B.F. Spencer, Jr., "Cumulant Minimization and Robust Control," in *Stochastic Theory and Adaptive Control,* Tyrone E. Duncan and Bozenna Pasik-Duncan, Editors. New York: Springer-Verlag Lecture Notes in Control and Information Sciences, pp. 411-425, 1992.

[16] M. K. Sain, C.-H. Won, and B. F. Spencer, Jr., "Cumulants in Risk-Sensitive Control: The Full-State-Feedback Cost Variance Case," *Proceedings IEEE Conference on Decision and Control,* New Orleans, pp. 1036-1041, December 1995.

[17] B. F. Spencer, Jr., S. J. Dyke, and H. S. Deoskar "First Generation Benchmark," Special Issue of *Earthquake Engineering and Structural Dynamics,* Vol. 27, No. 11, pp 1127-1139, 1998.

[18] C.-H. Won, "Cost Variance Control and Risk Sensitivity," M. S. Thesis, Department of Electrical Engineering, University of Notre Dame, 1992.

[19] C.-H. Won, M. K. Sain, and B. F. Spencer, Jr., "Active Structural Control: A Risk Sensitive Approach," in *Dynamics and Control of Large Structures,* Proceedings Ninth Virginia Polytechnic Institute and State University Symposium, Blacksburg, Virginia, pp. 155-166, May 1993.

[20] C.-H. Won, M. K. Sain, and B. F. Spencer, Jr., "Risk-Sensitive Structural Control Strategies," in *Computational Stochastic Mechanics,* P. Spanos, Editor. Rotterdam: A. A. Balkema, 1995, pp. 203-211. Proceedings Second International Conference on Computational Mechanics, Athens, Greece, June 1994.

[21] C.-H. Won, "Cost Cumulants in Risk-Sensitive and Minimal Cost Variance Control," *Ph.D. Dissertation,* Department of Electrical Engineering, University of Notre Dame, Notre Dame, Indiana, November 1994.

[22] C.-H. Won, M. K. Sain, and B. F. Spencer, Jr., "Performance and Stability Characteristics of Risk-Sensitive Controlled Structures under Seismic Disturbances," *Proceedings American Control Conference,* Seattle, Washington, pp. 1926-1930, June 1995.

[23] C.-H. Won, M. K. Sain, and B. F. Spencer, Jr.,"Relationships in Modern Stochastic Control: Risk-Sensitive and Minimal Cost Variance Control," *Proceedings Allerton Conference on Communication, Control, and Computing,* pp. 343-352, 1996.

[24] C.-H. Won, M. K. Sain, and B. F. Spencer, Jr.,"Interpretations of Risk-Sensitivity in Dynamic Optimization of Circuits and Systems," *Proceedings IEEE Asia-Pacific Conference on Circuits and Systems,* Seoul, Korea, pp. 191-194, November 1996.

[25] C.-H. Won, M. K. Sain, and B. F. Spencer, Jr., "Connections and Advances in Risk-Sensitive and MCV Stochastic Control," *Proceedings Second Asia-Pacific Control Conference,* Seoul, Korea, vol. III, pp. 59-62, July 1997.

Evaluation Criteria	LQG		2-Cumulant Controller A: $\mu_3 = 0$				2-Cumulant Controller B: $\mu_2 = 0$			
	$\mu_2 = \mu_3 = 0$		$\mu_2 = 0.0054$		$\mu_2 = 0.0072$		$\mu_3 = 9e-6$		$\mu_3 = 12e-6$	
J1	0.283977		0.213730 (−24.7)		0.185279 (−34.8)		0.229345 (−19.2)		0.200800 (−29.3)	
J2	0.439650		0.326080 (−25.8)		0.281283 (−36.0)		0.351275 (−20.1)		0.305196 (−30.6)	
J3	0.511421		0.722291 (41.2)		0.844314 (65.1)		0.666474 (30.3)		0.773928 (51.3)	
J4	0.512507		0.717216 (39.9)		0.834101 (62.7)		0.663337 (29.4)		0.766856 (49.6)	
J5	0.62668		0.757963 (20.9)		0.855152 (26.5)		0.716358 (14.3)		0.797899 (27.3)	
J6	0.402224	**0.455587**	0.342076 (−15.0)	**0.387583** (−14.9)	0.304935 (−24.2)	**0.374867** (−17.7)	0.359205 (−10.7)	**0.392822** (−13.8)	0.326438 (−18.8)	**0.382356** (−16.1)
J7	0.636272	**0.710194**	0.527572 (−17.1)	**0.673830** (−5.1)	0.457177 (−28.1)	**0.647909** (−8.8)	0.558401 (−12.2)	**0.685373** (−3.50)	0.498535 (−21.6)	**0.664170** (−6.48)
J8	0.593982	**0.667990**	0.999150 (68.2)	**1.213118** (81.6)	1.249476 (110.4)	**1.657354** (148.1)	0.878351 (47.9)	**1.032507** (54.6)	1.108668 (86.7)	**1.393689** (108.6)
J9	0.605380	**0.775300**	1.017308 (68.0)	**1.231123** (58.8)	1.251880 (106.8)	**1.575507** (103.2)	0.904633 (49.4)	**1.092183** (40.9)	1.119858 (85.0)	**1.368040** (76.5)
J10	0.955936	**1.336047**	1.050937 (9.9)	**1.454190** (8.8)	1.171717 (22.6)	**1.634610** (22.3)	1.003763 (5.00)	**1.391516** (4.15)	1.107707 (15.9)	**1.532302** (14.7)
	ElCentro	Hachinohe	ElCentro	Hachinohe	ElCentro	Hachinohe	ElCentro	Hachinohe	ElCentro	Hachinohe
$uRMS/uMAX$ $1.0\,V/3.0\,V$	0.143033/0.525495		0.222461/0.943184 (55.5)/(79.5)		0.270429/1.178174 (89.1)/(124.2)		0.200835/0.830423 (40.4)/(58.0)		0.242734/1.045702 (69.7)/(99.0)	
$amRMS/amMAX$ $2.0\,g/6.0\,g$	1.121766/4.827479		1.356753/5.307234 (20.9)/(9.9)		1.530722/5.917169 (36.5)/(22.6)		1.282281/5.069002 (14.3)/(5.00)		1.428239/5.593922 (27.3)/(15.9)	
$xmRMS/xmMAX$ $3.0\,cm/9.0\,cm$	0.669962/2.001718		0.946201/3.367136 (41.2)/(68.2)		1.106051/4.210734 (65.1)/(110.4)		0.873081/2.960045 (30.3)/(47.9)		1.013846/3.736212 (51.3)/(86.7)	

Table 1: Benchmark Evaluation Results

Evaluation	*LQG*		3-Cumulant Controller C: $\mu_3 = 2.55e-6$				2-Cumulant Controller D: $\mu_2 = 0.0054$			
Criteria	$\mu_2 = \mu_3 = 0$		$\mu_2 = 0.004$		$\mu_2 = 0.006$		$\Delta W = 20\%$		$\Delta W = 30\%$	
J1	0.283977		0.215379 (-24.2)		0.182495 (-35.7)		0.196580 (-30.8)		0.188076 (-33.8)	
J2	0.439650		0.328739 (-25.2)		0.277489 (-36.9)		0.298395 (-32.1)		0.285109 (-35.2)	
J3	0.511421		0.716061 (40.0)		0.858036 (67.8)		0.792143 (54.9)		0.830886 (62.5)	
J4	0.512507		0.711214 (38.8)		0.847162 (65.3)		0.784292 (53.0)		0.821307 (60.3)	
J5	0.626685		0.753140 (20.2)		0.866533 (38.3)		0.812828 (29.7)		0.844208 (34.7)	
J6	0.402224	**0.455587**	0.343935 (-14.5)	**0.388184** (-14.8)	0.300766 (-25.2)	**0.373302** (-18.1)	0.320881 (-20.2)	**0.380493** (-16.5)	0.309037 (-23.2)	**0.376334** (-17.4)
J7	0.636272	**0.710194**	0.530859 (-16.6)	**0.675588** (-4.87)	0.449204 (-29.4)	**0.646153** (-9.02)	0.488002 (-23.3)	**0.660106** (-7.05)	0.465069 (-26.9)	**0.650108** (-8.46)
J8	0.593982	**0.667990**	0.985711 (65.9)	**1.192009** (78.4)	1.275885 (114.8)	**1.711216** (156.2)	1.145927 (92.9)	**1.459557** (118.5)	1.223584 (106.0)	**1.605542** (140.4)
J9	0.605380	**0.775300**	1.004786 (66.0)	**1.212663** (56.4)	1.27696 (110.9)	**1.617706** (108.7)	1.154461 (90.7)	**1.421172** (83.3)	1.227352 (102.7)	**1.533220** (97.8)
J10	0.955936	**1.336047**	1.052682 (10.1)	**1.459211** (9.21)	1.185429 (24.0)	**1.649933** (23.5)	1.129457 (18.2)	**1.557955** (16.6)	1.153418 (20.7)	**1.605238** (20.1)
	ElCentro	Hachinohe	ElCentro	Hachinohe	ElCentro	Hachinohe	ElCentro	Hachinohe	ElCentro	Hachinohe
$uRMS/uMAX$ $1.0\,V/3.0\,V$	0.143033/0.525495		0.220037/0.930804 (53.8)/(77.1)		0.275849/1.202976 (92.9)/(128.9)		0.249862/1.080636 (74.7)/(105.6)		0.265129/1.153487 (85.4)/(119.5)	
$amRMS/amMAX$ $2.0\,g/6.0\,g$	1.121766/4.827479		1.348120/5.316043 (20.2)/(10.1)		1.551094/5.986418 (38.3)/(24.0)		1.454962/5.703756 (29.7)/(18.2)		1.511133/5.824760 (34.7)/(20.7)	
$xmRMS/xmMAX$ $3.0\,cm/9.0\,cm$	0.669962/2.001718		0.938040/3.321845 (40.0)/(65.9)		1.124027/4.299734 (67.8)/(114.8)		1.037707/3.861773 (54.9)/(92.9)		1.088460/4.123478 (62.5)/(106.0)	

Table 2: Benchmark Evaluation Results

Risk-Sensitive Decision-Theoretic Troubleshooting

Mark A. Shayman *
Electrical & Computer Eng. Dept. and
Institute for Systems Research
University of Maryland
College Park, MD 20742
shayman@eng.umd.edu

Emmanuel Fernández-Gaucherand †
Systems & Industrial Engineering Dept.
University of Arizona
Tucson, AZ 85721
emmanuel@sie.arizona.edu

Abstract

We study the problem of sequencing tests for the discovery of a faulty component, e.g., in a telecommunications network, where there is a random cost associated with testing a component. A novel feature in our approach is that a risk-sensitive performance criterion is used in order to rank different competing schedules.

1 Introduction

Consider the problem of fault management for communication networks. As networks increase in size and complexity, the frequency of faults increases. At the same time, the tremendous increase in network speed means that the cost per unit time of failures is much greater than in the recent past. Network faults may be difficult to predict or diagnose, and may result in huge costs.

Our focus in this paper is on *sequential* search techniques for fault detection and diagnosis. Thus, it is concerned with the optimal scheduling of monitoring/testing tasks for a single intelligent agent. We seek to obtain rules that lead to schedules exhibiting an "optimal" compromise between acceptable variance and expected costs. This is achieved via the use of an exponential utility function, leading to an optimization process that yields risk (i.e., variance) sensitive scheduling rules; see [2], [5], [7], [15], [23], [27], [36].

---

*Research Supported in part by NSF grant ECS-9626399 and DoD contract MDA90497C3015
†Research supported in part by NSF grant INT-9602939 and by United Engineering Foundation grant 00/ER-09

1.1 The Fault Management Process

The fault management process can be viewed as a hierarchical process, consisting of two phases. In the first phase, the agent searches its domain for evidence of anomalous behavior. If evidence of such behavior is detected in a subdomain, a second phase begins in which the network element(s) in the subdomain are tested to isolate the specific fault. The following generic search problem corresponds to the phase one problem: Given a set of n components which fail *independently*, each of which may be tested with random testing cost to determine whether or not it is faulty, determine the "optimal" order in which the components should be tested. We refer to this as the *independent fault* (INF) problem. The phase two problem corresponds to a different generic search problem: Given a set of n components *with at most one in failure*, determine the "optimal" order in which the components should be tested. We refer to this as the *mutually exclusive fault* (MEF) problem.

When the objective is to minimize the expected sum of the testing costs, both INF and MEF are classical problems that have been investigated over several decades. Various aspects of the INF problem have been considered by (among others) Bellman [4], [25] Dean [8], Kadane [20], Joyce [19], Garey [10], Simon and Kadane [32], Kadane and Simon [21], Monma and Sidney [26]. Work related to the MEF problem has been reported by (among others) Bellman [4], Staroverov [33], Greenberg [12], Matula [24], Black [6], Denby [9], Horn [14], Adolphson and Hu [1], Sidney [30], Stone [34], Hall [13].

In this paper we concentrate on the MEF problem, when there are no constraints on the admissible testing sequence. A more comprehensive study is presented elsewhere [29], where both the INF and the MEF problems are treated, with and without constraints on the admissible testing sequence.

Consider the MEF problem and suppose that the probability p_i that each component i is faulty is known–from prior knowledge of the properties of the system together with any observations made before the commencement of testing. Each component i can be tested at a random cost C_i with a binary-valued result indicating whether or not the component is faulty. Suppose that $\{C_i\}_{i=1}^n$ are independent nonnegative random variables and are independent of $\{I_{F_i}\}_{i=1}^n$, where I_{F_i} is the indicator function of fault F_i. (Note that this assumes that the cost of testing component i is independent of whether or not it is faulty.) The components are tested in some sequential order until the (unique) faulty component is discovered at which time the testing terminates. Suppose the goal is to determine the order in which the tests are performed to minimize the expected value of the sum of the testing costs. Since any test for which the cost is 0 (w.p.1) would trivially be scheduled before any other test, there is no loss of generality in excluding the possibility that $C_i = 0$ (w.p.1) for any i. If there are no precedence constraints, an interchange argument [5] can be used to show that it is optimal to schedule the tests in order of increasing ratios $E[C_i]/p_i$. This has been referred to in the literature as the *C/P algorithm* [22]. To motivate our presentation in the sequel, we proceed with a simple derivation of the C/P algorithm: Let $q_i := 1 - p_i$, and assume that $(t_1, t_2, \ldots, t_k, i, j, \ldots, t_n)$ is an optimal schedule; the expected cost associated with this schedule is

$$E[C_{t_1} + q_{t_1} C_{t_2} + \ldots + (\Pi_{\ell=1}^k q_{t_\ell}) \cdot C_i + (\Pi_{\ell=1}^k q_{t_\ell}) \cdot q_i C_j + \ldots].$$

Comparing the optimal (minimal) cost above with that yielded by the (suboptimal) schedule obtained by interchanging the i and j tests, we obtain the necessary condition for optimality:

$$E[C_i] + q_i E[C_j] \leq E[C_j] + q_j E[C_i],$$

which is equivalent to the C/P rule:

$$\frac{E[C_i]}{p_i} \leq \frac{E[C_j]}{p_j}.$$

It is also easy to conclude that an optimal policy for this problem is an (open-loop) schedule, and hence the C/P rule is actually necessary and sufficient for optimality [5].

Huard and Lazar [17] have proposed using the the C/P algorithm in conjunction with Bayesian belief networks to determine testing strategies for communication network fault isolation. The belief networks represent domain knowledge and are used to determine the failure probabilities p_i; see also [16]. (They also discuss the use of dynamic programming for models in which the C/P-rule is not applicable.) Baras, Li, and Mykoniatis [3] have proposed a heuristic based upon which the C/P algorithm can be adapted to the case of multiple faults.

1.2 Optimality Criteria

In the literature mentioned previously, the objective is to minimize the average sum of the testing costs. This may make sense for diagnostic problems that will be repeated many times under the same conditions, i.e., with the same model. This is unlikely to be the case for the diagnosis of communication network faults. This concern leads us to consider the problem of optimal sequential fault diagnosis with a *risk-sensitive optimality criterion.* This is achieved via the use of an exponential utility function, leading to an optimization process that yields risk (i.e., variance) sensitive scheduling rules; see [2], [5], [15], [23], [27], [36].

We therefore assume that risk sensitivity is given by an exponential (dis-) utility function $U_\gamma(x) = (\text{sgn}\gamma)e^{\gamma x}$, $\gamma \in \Re$, $\gamma \neq 0$. The above utility function can be explained as modeling the behavior of a "decision maker" with a constant level of risk aversion, as given by the parameter γ [28], [23]. Another viewpoint is just to consider a Taylor expansion of $E[e^{\gamma C}]$ about $E[C]$, where C represents the sum of aggregated random costs. Up to a first approximation, the latter leads to an objective for optimization composed of the sum of expected costs (the standard or *risk-null* criterion) and the *variance of* C multiplied by the risk sensitivity parameter γ. Hence, if $\gamma > 0$ the decision maker is said to be *risk-averse* in that the objective for the optimization seeks to minimize both variance (uncertainty measuring risks involved) and the expected costs. If $\gamma < 0$, then variance is actually seen as a desirable feature, and the decision maker is said to be *risk-seeking.* Finally, if $\gamma = 0$, the standard expected costs criterion is recovered, which cannot distinguish between two scheduling rules with different variances but equal expected costs, a risk insensitive or *risk-null* situation.

In the sequel, for the MEF problem without constraints, we are able to obtain explicit characterization of the optimal testing sequence in the unconstrained case. At the root of our results is an interchange argument. That such an argument proves useful is somewhat surprising since scheduling problems that can be solved by interchange arguments in the risk-neutral case are not necessarily amenable to such solution in the case of risk-sensitive criterion [2].

2 Mutually Exclusive Faults Without Precedence Constraints

Let C_i be a random variable representing the cost of testing component i given that it is faulty, and let D_i be a random variable representing the cost of testing component i given that it is not faulty. The relationship between C_i and D_i–e.g., the relative magnitudes of $E[C_i]$ and $E[D_i]$ is application-dependent. We assume that conditional on the state of component i, the testing cost of component i is independent of the states and testing costs of all other components. We allow for the possibility that the true fault may not be one of the n hypothesized faults; thus $\sum_{i=1}^{n} p_i$ may be less than 1. If there is a faulty component, the testing sequence terminates when that component is tested; otherwise, every component is tested.

The exponential utility function used in our risk-sensitive optimality criterion also has some relation to scheduling problems which involve minimizing completion times, and when (exponential) discounting is employed, as illustrated in the following remark.

Remark 1 In the special case where $C_i = D_i$, $\forall i$ and $\gamma < 0$ (risk-seeking decision maker), the MEF problem without precedence constraints is mathematically equivalent to the total weighted exponential completion time problem [11, 26]. To obtain this equivalence, p_i and C_i are redefined to be the weight and processing time, respectively, for job i.

■

Let $\psi(i;\gamma) := E[e^{\gamma C_i}]$, and let $\overline{\psi}(i;\gamma) := E[e^{\gamma D_i}]$. Thus, $\psi(i;\gamma)$ and $\overline{\psi}(i;\gamma)$ are the moment generating functions of C_i and D_i, respectively. In the sequel, we generally suppress the dependence on γ.

Let $a = (a_1, \ldots, a_m)$ be a permutation of an m-element subset, denoted $\{a\}$, of $\{1, \ldots, n\}$. We refer to a as a schedule and indicate the number of elements in the schedule by $|a|$. If $|a| = n$, we refer to a as a complete schedule. For two (disjoint) schedules $a = (a_1, \ldots, a_m)$ and $b = (b_1, \ldots, b_\ell)$ such that $m + \ell \leq n$, we denote the concatenated schedule as

$$ab := (a_1, \ldots, a_m, b_1, \ldots, b_\ell).$$

To simplify notation, we will generally assume that $\gamma > 0$–i.e., the decision maker is risk-averse. However, we will indicate the analogous results for the case $\gamma < 0$ as well. Denote the random cost as

$$\mathcal{C}_i = I_{F_i} C_i + (1 - I_{F_i}) D_i,$$

and for $a = (a_1, \ldots, a_m)$, the utility of the aggregated cost is computed as

$$\mathcal{C}(a) = exp\{\gamma \sum_{i=1}^{m} \mathcal{C}_{a_i}\}. \tag{1}$$

In the event that $I_{F_{a_i}} = 1$, i.e., the a_i^{th} component is faulty, then

$$E[\mathcal{C}(a) | I_{F_{a_i}} = 1] = \psi(a_i) \prod_{j=1}^{i-1} \overline{\psi}(a_j). \tag{2}$$

Hence, for $|a| = m > 1$, the expected exponential risk-sensitive cost $V(a)$ for the schedule a is given by

$$V(a) = \sum_{i=1}^{m} p_{a_i}\psi(a_i)\prod_{j=1}^{i-1}\overline{\psi}(a_j) + (1 - \sum_{i=1}^{m} p_{a_i})\prod_{i=1}^{m}\overline{\psi}(a_i), \tag{3}$$

where the first term comes from (2) and the independence assumptions, and the second term accounts for the probability that none of the $|a| = m$ tested components are faulty. In the special case where $|a| = 1$ and $a = (i)$, we have

$$V(i) = p_i\psi(i) + (1 - p_i)\overline{\psi}(i). \tag{4}$$

If we want to emphasize the dependence on the risk-sensitivity parameter γ, we will use the notation $V(a;\gamma)$ instead of $V(a)$.

The following result plays a crucial role in the sequel.

Lemma 1 *Suppose that* $\{a\} = \{b\}$. *Then* $V(a) \leq V(b)$ *if and only if*

$$\sum_{i=1}^{m} p_{a_i}\psi(a_i)\prod_{j=1}^{i-1}\overline{\psi}(a_j) \leq \sum_{i=1}^{m} p_{b_i}\psi(b_i)\prod_{j=1}^{i-1}\overline{\psi}(b_j). \tag{5}$$

Proof: The result follows directly from (3). Let $m = |a| = |b|$. Since $\{a\} = \{b\}$,

$$(1 - \sum_{i=1}^{m} p_{a_i})\prod_{i=1}^{m}\overline{\psi}(a_i) = (1 - \sum_{i=1}^{m} p_{b_i})\prod_{i=1}^{m}\overline{\psi}(b_i). \tag{6}$$

■

Let

$$\hat{V}(a) := \sum_{i=1}^{m} p_{a_i}\psi(a_i)\prod_{j=1}^{i-1}\overline{\psi}(a_j). \tag{7}$$

As a consequence of Lemma 1, when comparing schedules containing the same set of elements, it is equivalent to base the comparison on the function $\hat{V}$ rather than on V. This greatly simplifies the analysis.

Let $\overline{\psi}(a) := \prod_{i=1}^{|a|}\overline{\psi}(a_i)$. It follows easily from the definitions that

$$\overline{\psi}(ab) = \overline{\psi}(a)\overline{\psi}(b), \tag{8}$$

$$\hat{V}(ab) = \hat{V}(a) + \overline{\psi}(a)\hat{V}(b). \tag{9}$$

The following proposition contains the key interchange argument used subsequently to obtain the optimal scheduling rules.

Proposition 1 *If* $|i| = |j| = 1$, *then*

$$\hat{V}(aijd) \leq \hat{V}(ajid) \Leftrightarrow \frac{\overline{\psi}(i) - 1}{p_i\psi(i)} \leq \frac{\overline{\psi}(j) - 1}{p_j\psi(j)}. \tag{10}$$

Proof: Iterating (9), we obtain

$$\hat{V}(abcd) = \hat{V}(a) + \overline{\psi}(a)\hat{V}(b) + \overline{\psi}(a)\overline{\psi}(b)\hat{V}(c) + \overline{\psi}(a)\overline{\psi}(b)\overline{\psi}(c)\hat{V}(d). \tag{11}$$

Hence, using (8) and (11), we obtain

$$\hat{V}(abcd) - \hat{V}(acbd) = \overline{\psi}(a)\{\hat{V}(c)[\overline{\psi}(b) - 1] - \hat{V}(b)[\overline{\psi}(c) - 1]\}, \tag{12}$$

and thus

$$\hat{V}(abcd) \leq \hat{V}(acbd) \quad \Leftrightarrow \quad \frac{\hat{V}(c)}{\psi(c)-1} \leq \frac{\hat{V}(b)}{\overline{\psi}(b)-1} \quad \Leftrightarrow \quad \frac{\overline{\psi}(b)-1}{\hat{V}(b)} \leq \frac{\overline{\psi}(c)-1}{\hat{V}(c)},$$

from which the result follows with $b = i$ and $c = j$. ■

Remark 2 That an interchange argument leads to optimal "policies" in the present situation is clearly true, since no information gathering is done while a schedule is being implemented, and thus the optimal policy is an open-loop schedule [5]. This observation leads to the following scheduling rule.

Theorem 1 *Let $\gamma \neq 0$. For the MEF problem with no precedence constraints, a complete schedule $t = (t_1, \ldots, t_n)$ is optimal if and only if*

$$\frac{\operatorname{sgn}(\gamma)(\overline{\psi}(t_i)-1)}{p_{t_i}\psi(t_i)} \leq \frac{\operatorname{sgn}(\gamma)(\overline{\psi}(t_j)-1)}{p_{t_j}\psi(t_j)}, \tag{13}$$

whenever $i < j$.

Proof: For the case $\gamma > 0$, the result follows immediately by applying an interchange argument, based on Proposition 1. The analogous result for the case $\gamma < 0$ is derived similarly. Note that if $\gamma < 0$, then $\psi(i) < 1$ and $\overline{\psi}(i) < 1$. ■

Thus, from Theorem 1, we conclude that the components should be tested in increasing order of the ratios

$$\frac{\operatorname{sgn}(\gamma)(E[e^{\gamma D_i}]-1)}{p_i E[e^{\gamma C_i}]}.$$

The following result shows that the classical ratio test for the risk-neutral problem is recovered when the absolute value of the risk-sensitivity parameter is small.

Corollary 1 *Suppose that the ratios $\{E[D_i]/p_i\}_{i=1}^{n}$ are distinct. Then there exists $\epsilon > 0$ such that if $|\gamma| < \epsilon$ then there is a unique optimal complete schedule $t = (t_1, \ldots, t_n)$ for the MEF problem, and it is specified by the condition that whenever $i < j$, then*

$$\frac{E[D_{t_i}]}{p_{t_i}} < \frac{E[D_{t_j}]}{p_{t_j}}. \tag{14}$$

Proof: For small $|\gamma|$, $\psi(i) \approx 1 + \gamma E[C_i]$ and $\overline{\psi}(i) \approx 1 + \gamma E[D_i]$. Using these approximations, the assertion follows from Theorem 1. ■

Remark 3 From Corollary 1 we see that if the risk-sensitivity parameter is sufficiently small, the costs $\{C_i\}$ associated with *positive* tests are irrelevant to the determination of the optimal testing order. In particular this is the case in the risk-neutral problem–i.e., in the limit as $\gamma \to 0$. However, from Theorem 1, it follows that for the general risk-sensitive problem the costs associated with positive tests as well as costs associated with negative tests are relevant to the optimal testing order, a rather important distinction with respect to the risk-neutral case, i.e., the C/P algorithm.

■

Next, we consider the structure of the optimal schedule for $|\gamma| >> 0$. For this discussion, we make the simplifying assumption that for each i, the random variables C_i and D_i are constants with values c_i and d_i, respectively.

Corollary 2 *Suppose that* $C_i = c_i$ *and* $D_i = d_i$, $\forall i$. *Suppose that the complete schedule* $t = (t_1, \ldots, t_n)$ *is such that if* $i < j$, *then one of the following conditions holds:*

(a) $d_{t_i} - c_{t_i} < d_{t_j} - c_{t_j}$.

(b) $d_{t_i} - c_{t_i} = d_{t_j} - c_{t_j}$ *and* $p_{t_j} < p_{t_i}$.

(c) $d_{t_i} - c_{t_i} = d_{t_j} - c_{t_j}$ *and* $p_{t_j} = p_{t_i}$ *and* $c_{t_i} < c_{t_j}$.

Then for all sufficiently large $\gamma > 0$, t *is the unique optimal complete schedule for the MEF problem.*

Proof: From Proposition 1 it follows that $\hat{V}(aijb) < \hat{V}(ajib)$ if and only if

$$p_j e^{\gamma c_j}(e^{\gamma d_i} - 1) < p_i e^{\gamma c_i}(e^{\gamma d_j} - 1). \tag{15}$$

By considering (15) for $\gamma >> 0$ and using an interchange argument, the result follows. ■

In particular, this result shows that if the differences $\{d_i - c_i\}$ are distinct, then a very risk-averse decision maker should order the tests so that the differences between the cost of a negative test and positive test are increasing. On the other hand, if there are two tests for which these differences are equal, the test with higher probability of success (i.e., testing the component with higher failure rate) should be scheduled first. If both the differences and probabilities are equal, the test with the smaller cost should be scheduled first. In the special case in which the cost of negative tests is the same as that of positive tests, and in which the success probabilities are all distinct, the tests should be scheduled in order of decreasing success probability. The behavior exhibited by a very risk-averse decision maker is therefore "myopic."

Corollary 3 *Suppose that* $C_i = c_i$ *and* $D_i = d_i$, $\forall i$. *Suppose that the complete schedule* $t = (t_1, \ldots, t_n)$ *is such that if* $i < j$, *then one of the following conditions holds:*

(a) $c_{t_i} < c_{t_j}$.

(b) $c_{t_i} = c_{t_j}$ *and* $p_{t_j} < p_{t_i}$.

(c) $c_{t_i} = c_{t_j}$ *and* $p_{t_j} = p_{t_i}$ *and* $d_{t_i} < d_{t_j}$.

Then for all $\gamma < 0$ *with* $|\gamma|$ *sufficiently large,* t *is the unique optimal complete schedule for the MEF problem.*

Proof: Follows easily from Proposition 1 and Theorem 1, e.g., reversing the inequality in (15). ■

Example 1 Suppose there are three components with fault probabilities $p_1 = 0.3$, $p_2 = 0.4$, $p_3 = 0.2$. Note that there remains a probability of 0.1 that the failure is not in any of the three suspected components. Suppose that the costs of negative tests are $D_1 = 2$, $D_2 = 3$, $D_3 = 2$ and the costs of positive tests are $C_1 = 1$, $C_2 = 2$, $C_3 = 3$. Thus negative tests are more costly than positive tests for components 1 and 2, but the opposite is true for component 3. Thus, it is more costly to exonerate components 1 and 2 than it is to confirm failure and repair them; the opposite is true for component 3. For example, component 3 might have a substantial repair cost.

Since $D_1/p_1 = 6.67$, $D_2/p_2 = 7.50$, $D_3/p_3 = 10.00$, it follows from Corollary 1 that for small $|\gamma|$, the optimal schedule is $(1, 2, 3)$. In particular this is the case in the limit as $\gamma \rightarrow 0$–i.e., the risk-neutral case. Note that the costs associated with positive tests, and hence repair costs, have no bearing on this conclusion.

Since, $c_1 < c_2 < c_3$, it follows from Corollary 3 that for sufficiently negative values of γ–i.e., for a sufficiently risk-seeking decision maker–the optimal schedule is $(1, 2, 3)$. In fact, numerical investigation confirms that this is the optimal schedule for all $\gamma \leq 0$.

Since $d_3 - c_3 < d_1 - c_1 = d_2 - c_2$ and $p_1 < p_2$, it follows from Corollary 2 that for sufficiently positive values of γ–i.e., for a sufficiently risk-averse decision maker–the optimal schedule is $(3, 2, 1)$. In fact, numerical investigation based on Theorem 1 reveals that the optimal schedule is $(1, 2, 3)$ for $-\infty < \gamma \leq 0.1869$, $(1, 3, 2)$ for $[0.1870, 0.2027]$, $(3, 1, 2)$ for $[0.2028, 0.2644]$, and $(3, 2, 1)$ for $[0.2645, \infty)$.

■

References

[1] D. Adolphson and T. C. Hu. Optimal linear ordering. *SIAM J. Appl. Math.*, 25:403–423, 1973.

[2] G. Avila-Godoy and E. Fernandez-Gaucherand. Exponential risk-sensitive optimal scheduling. In *Proceedings of the 36th CDC*, pages 3958–3963, San Diego, CA, December 1997.

[3] J. Baras, H. Li, and G. Mykoniatis. Integrated, distributed fault management for communication networks. Technical Report 98-10, Center for Satellite and Hybrid Communication Networks, University of Maryland, College Park, MD, 1998.

[4] R. Bellman. *Dynamic Programming.* Princeton University Press, Princeton, New Jersey, 1957.

[5] D. Bertsekas. *Dynamic Programming and Optimal Control, Volume 1.* Athena Scientific, Belmont, MA, 1995.

[6] W. Black. Discrete sequential search. *Information and Control*, 8:156–162, 1965.

[7] R. Cavazos-Cadena and E. Fernández-Gaucherand. Controlled markov chains with risk-sensitive criteria: average cost, optimality equations, and optimal solutions. *Mathematical Methods of Operations Research*, 49:299–324, 1999.

[8] B. V. Dean. Stochastic networks in research planning. In Yovits et al., editor, *Research Program Effectiveness*, chapter 12. Gordon and Breach, New York, 1966.

[9] D. C. Denby. Minimum downtime as a function of reliability and priority in a component repair. *Journal of Industrial Engineering*, 18:436–439, 1967.

[10] M. R. Garey. Optimal sequencing with precedence constraints. *Discrete Mathematics*, 4:37–56, 1973.

[11] K.D. Glazebrook and J.C. Gittens. On single-machine scheduling under precedence constraints and linear or discounted costs. Technical report, Mathematical Institute, University of Oxford, 1979.

[12] H. Greenberg. Optimum test procedure under stress. *Operations Research*, 12:689–692, 1964.

[13] G. J. Hall. Sequential search with random overlook probabilities. *Annals of Statistics*, 4:807–816, 1976.

[14] W. A. Horn. Single-machine job sequencing with treelike precedence ordering and linear delay penalties. *SIAM J. Appl. Math.*, 23:189–202, 1972.

[15] A. R. Howard and J. E. Matheson. Risk-sensitive markov decision processes. *Management Sciences*, 18:356–369, 1972.

[16] J-F. Huard. Probabilistic reasoning for fault management on xunet. Technical report, Summer Report, AT&T Bell Labs, Murray Hill, NJ, 1994.

[17] J-F. Huard and A.A. Lazar. Fault isolation based on decision-theoretic troubleshooting. Technical Report 442-96-08, Center for Telecommunications Research, Columbia University, New York, NY, 1996.

[18] F.V. Jensen. *An Introduction to Bayesian Networks.* Springer, New York, 1996.

[19] W. B. Joyce. Organization of unsuccessful r&d programs. *IEEE Transactions on Engineering Management*, 18:57–65, 1971.

[20] J. B. Kadane. Quiz show problems. *Journal of Mathematical Analysis and Applications*, 27:609–623, 1969.

[21] J. B. Kadane and H. A. Simon. Optimal strategies for a class of constrained sequential problems. *Annals of Statistics*, 5(2):237–255, 1977.

[22] J. Kalagnanam and M. Henrion. A comparison of decision analysis and expert rules for sequential diagnosis. In *Uncertainty in Artificial Intelligence 4*, pages 271–281. Elsevier, 1990.

[23] S. I. Marcus, E. Fernández-Gaucherand, D. Hernández-Hernández, S. Coraluppi, and P. Fard. Risk sensitive markov decision processes. In C. I. Byrnes, B. N. Datta, D. S. Gilliam, and C. F. Martin, editors, *Systems & Control in the Twenty-First Century*, pages 263–279. Birkhauser, 1996.

[24] D. Matula. A periodic optimal search. *Amer. Math. Monthly*, 71:15–21, 1964.

[25] L. G. Mitten. An analytic solution to the least cost testing sequence problem. *Journal of Industrial Engineering*, 11:17, 1960.

[26] C. L. Monma and J. B. Sidney. Optimal sequencing via modular decomposition: Characterization of sequencing functions. *Mathematics of Operations Research*, 12(1):22–31, 1987.

[27] M. Pinedo. *Scheduling: Theory, Algorithms, and Systems.* Prentice Hall, Englewood Cliffs, NJ, 1995.

[28] J. W. Pratt. Risk aversion in the small and in the large. *Econometrica*, 32:122–136, 1964.

[29] M. A. Shayman and E. Fernández-Gaucherand. Risk-sensitive decision-theoretic troubleshooting. Technical Report, July 8, 1999.

[30] J. B. Sidney. Decomposition algorithms for single-machine sequencing with preference relations and deferral costs. *Operations Research*, 23:283–298, 1975.

[31] J. B. Sidney. A decomposition algorithm for sequencing with general precedence constraints. *Mathematics of Operations Research*, 6:190–204, 1981.

[32] H. A. Simon and J. B. Kadane. Optimal problem-solving search: All-or-none solutions. *Artificial Intelligence*, 6:235–247, 1975.

[33] O. Staroverov. On a searching problem. *Theory Probability Appl.*, 8:184–187, 1963.

[34] L. D. Stone. *Theory of Optimal Search.* Academic Press, New York, 1975.

[35] A. Tanenbaum. *Computer Networks.* Prentice Hall PTR, Upper Saddle River, NJ, 1996.

[36] P. Whittle. *Risk-sensitive Optimal Control.* Wiley, New York, 1990.

H_∞ Control for Mixed Disturbance Rejection*

J.C.Luo and E.B.Lee
jcluo@ece.umn.edu and eblee@ece.umn.edu
Department of Electrical and Computer Engineering
University of Minnesota, 200 Union Street,
Minneapolis, MN 55455

Abstract

In this paper we study an optimal H_∞ control problem related to the rejection of mixed disturbances for a class of linear systems. One of the disturbances is impulsive; its structure is given with action that is affine $v = \sum_{i=1}^{k} c_i\delta(t-t_i)$. In this case the solution of systems differential equations is a discontinuous function. After changing the original task into $k+1$ auxiliary combined optimal tasks removing the jump points, we derive necessary and sufficient conditions for the design of an optimal H_∞ controller.

Key words: Impulse control, minimax, LQR, H_∞ control.

1 Introduction

H_∞ control for impulsive disturbances was studied in reference [2] for a particular case by assuming that the linear system is a constant one and the impulsive disturbances are to fire within a constant time reset interval (that is $v = \sum_{i=1}^{\infty} c_i\delta(t-iT)$, here $T > 0$ is the constant reset time).

Here, we consider a more general case. The system has a mixture of two kinds of disturbance inputs, one of them is the impulsive disturbance input $v = \sum_{i=1}^{k} c_i\delta(t-t_i)$ (Notice here $c_i,\ i = 1, 2, ..k$ and $t_1\ t_2\ ...t_k$ are parameters to be selected, $\delta(t)$ is the Dirac delta function) and the other is a normal L_2 function. We developed the necessary and sufficient conditions for the design of H_∞ controller with this fixed disturbance action.

This paper is organized as follows. Section two contains the formulation and background material. Main results are given in section three. Conclusions are in section four.

2 Controller Design Task

Consider as [2] a formulation given in terms of the specification of system dynamics, performance index, and general assumptions. Let l_2 denote the Hilbert space of square-summable complex-valued sequences $\{x_k, k = 1, 2, 3, ...\}$ and L_2 denote the Hilbert space of all square integrable functions $x(t)$ defined on $[t_0, t_f]$.

---

*Research supported by the NSF grant DMS-9308007 and by the Vincentine Hermes-Luh endowment.

We consider linear systems with finite-dimensional state space as given by differential equations on $[t_0, t_f]$

$$\begin{aligned}
\dot{x}(t) &= A(t)x(t) + B_1(t)u(t) + B_2(t)v + B_3(t)w(t), \\
\text{with outputs } z(t) &= C_1(t)x(t) + D_1(t)u(t), \\
\text{and } y(t) &= C_2(t)x(t). \\
\text{Here } v(t) &= \sum_{i=1}^{k} c_i\delta(t-t_i) \text{ with to be determined parameters } c_i \text{ and } t_i \\
\text{and } w(t) &\in \|w\|_{L^2[t_0,t_f]} \text{ are the disturbances.} \\
x(t_{0-}) &= x_0 \qquad (1)
\end{aligned}$$

As usual $x \in R^n$, $u \in R^m$, $v \in R^k$, $y \in R^l$, $w \in R^{n1}$, $u(t)$ is vector of control variables which are measurable on $t \in [t_0, t_f]$, $w(t)$ is a piecewise continuous disturbance with $w(t) \in \|w\|_{L^2[t_0,t_f]}$ and A, B_1, B_2 , B_3, C_1, C_2 and D_1 all are proper dimensioned matrices.

The control task is to design a stabilizing feedback controller $u = u(y)$ to attenuate effects of the impulsive disturbance $v(t)$ on the controlled output $z(t)$.

Define a minimax performance measure as

$$J = \sup_{-u,\, v,\, w} \frac{\|z\|^2_{L_2(t_0,t_f)}}{\|v\|^2_{l_2}+\|w\|^2_{L^2[t_0,t_f]}} \tag{2}$$

The control objective is to find a controller $c_f \in K$ where K is a fixed class of controllers to solve the following task. Given a real $\gamma > 0$, try to find a $c_f \in K$ such that following inequality holds,

$$J(c_f) < \gamma^2 \tag{3}$$

for all possible $v \in l_2$, $w \in \|w\|_{L^2[t_0,t_f]}$ subject to the system (1). If such a controller c_f exists, we call it a γ -level disturbance attenuation controller.

3 Main Results

Assume $x(t_{0-}) = 0$, then change original task into $k+1$ combined tasks as in [3]:

$$\begin{aligned}
\dot{x}_i &= A(t)x_i + B_1(t)u_i + B_3(t)w_i, \\
z_i(t) &= C_1(t)x_i + D_1(t)u_i, \\
t &\in [t_{i-1}, t_i], \\
x_{i+1}(t_i) &= x_i(t_i) + B_2(t_i)c_i, \\
x_1(0) &= x(t_{0-}) = 0, \\
i &= 1, 2, 3 \ldots k+1 \\
J &= \sup_{-u,\, v,\, w} \frac{\|z\|^2_{L_2(t_0,t_f)}}{\|v\|^2_{l_2}+\|w\|^2_{L^2[t_0,t_f]}} = \sum_{i=1}^{k+1} \sup_{-u_i, c_i, t_i, w_i} \frac{\|z\|^2_{L_2(t_{i-1},t_i)}}{\|v\|^2_{l_2}+\|w_i\|^2_{L^2[t_{i-1},t_i]}}, \\
t_{n+1} &= t_f. \qquad (4)
\end{aligned}$$

We assume that all state variables are measured ($C_2(t) = I$), and also the following two standard assumptions obtain (i) (A, B_1) is controllable, (C_1, A) is observable and (ii) $D_1^T[C_1\, D_1] = [0\, I]$. Then we get:

Theorem 3.1 Consider the system (1) with the performance criterion (2) and $v = \sum_{i=1}^{k} c_i\delta(t-t_i)$. Suppose that assumptions (i) and (ii) hold. Then if there exists an optimal controller there is $c_f \in K$ such that $J(c_f) < \gamma^2$ if and only if there exist $P_i(t) \geq 0$, $i = 1, 2, \ldots k+1$ such that:

$$\begin{array}{rcl}
\dot{x}_i & = & A(t)x_i + B_1(t)u_i^* + B_3(t)w_i^*, \\
z_i(t) & = & C_1(t)x_i + D_1(t)u_i^*, \\
x_{i+1}(t_i) & = & x_i(t_i) + B_2(t_i)c_i, \\
x_1(0) & = & x(t_{0-}) = 0, \\
\dot{P}_i(t) = -A^T(t)P_i(t) & - & P_i(t)A(t) + P_i(t)(B_1(t)B_1^T(t) - \gamma^{-2}B_3(t)B_3^T(t))P_i(t) \\
 & - & C_1^T(t)C_1(t), \ t \in [t_{i-1}, t_i], \\
P_{k+1}(t_f) & = & 0, \\
P_i(t_i) = P_{i+1}(t_i) & + & (B_2^T(t_i)P_{i+1}(t_i))^T(\gamma^2 I \\
 & - & B_2^T(t_i)P_{i+1}(t_i)B_2(t_i))^{-1}B_2^T(t_i)P_{i+1}(t_i), \\
u_i^*(t) & = & -B_1^T(t)P_i(t)x_i(t) \\
w_i^*(t) & = & \gamma^{-2}B_3^T(t)P_i(t)x_i(t), \\
\text{all possible} \quad t_i's & & \text{are decided by} \\
x_i^T(t_i)C_1^T(t)C_1(t)x_i(t_i) & - & x_{i+1}^T(t_i)C_1^T(t)C_1(t)x_{i+1}(t_i) + 2x_i^T(t_i)P_i(t_i)\dfrac{dB_2(t)}{dt}c_i\,|_{t=t_i} = 0, \\
t_1 & = & t_0, \ t_k = t_f, \\
c_i^* & = & \gamma^{-2}B_2^T(t_i)P_i(t_i)x_i(t_i), \\
i & = & 1, 2, 3, ..k
\end{array} \tag{5}$$

Proof: Omitted here.

Next we will consider the situation when $t_f \to \infty$ and assume A, B_1, B_2, C_1 and D_1 are all constant matrices. In this case, we obtain:

Theorem 3.2 Consider the system (1) (assume A, B_1, B_2, C_1 and D_1 are all constant matrices) with the performance criterion (2) when $t_f \to \infty$ and $v = \sum_i c_i\delta(t-t_i)$. Suppose that assumptions (i) and (ii) hold. Then for the controller $c_f \in K$ such that $J(c_f) < \gamma^2$ get necessary and sufficient conditions as follows. Let there be $P_i(t) > 0$, $i = 1, 2, \ldots k+1$ such that:

$$\begin{array}{rcl}
\dot{x}_i & = & Ax_i + B_1u_i + B_3w_i, \\
z_i(t) & = & C_1x_i + D_1u_i, \\
x_1(0) & = & x(t_{0-}) = 0, \\
\dot{P}_i(t) = -A^TP_i(t) - P_i(t)A & + & P_i(t)(B_1B_1^T(t) - \alpha^{-2}B_3B_3^T)P_i(t) - C_1^TC_1, \\
t & \in & [t_{i-1}, t_i], \\
P_i(t_i) = P_{i+1}(t_i) & + & (B_2^TP_{i+1}(t_i))^T(\alpha^2 I - B_2^TP_{i+1}(t_i)B_2)^{-1}B_2^TP_{i+1}(t_i), \\
u_i(t) & = & -B_1^TP_i(t)x_i(t) \\
w_i(t) & = & \alpha^{-2}B_3^TP_i(t)x_i(t) \\
\text{all possible} \quad t_i's & & \text{are decided by} \\
x_i^T(t_i)C_1^TC_1x_i(t_i) & - & x_{i+1}^T(t_i)C_1^TC_1x_{i+1}(t_i) = 0, \\
t_1 & = & t_0, \\
c_i & = & \alpha^{-2}B_2^TP_i(t_i)x_i(t_i),
\end{array}$$

$$0<\alpha < \gamma,$$

Further more $P_i(t_i)$ and α should be selected to let the hybrid system stable

$$\begin{aligned} x_{i+1}(t_i) &= (I + B_2\alpha^{-2}B_2^T P_i(t_i))\Phi(t_i, t_{i-1})x_i(t_{i-1}), \\ i &= 1, 2, 3, ..k... \end{aligned} \tag{6}$$

here $\Phi(t_i, t_{i-1})$ is the state transition matrix of the differential equation $\dot{x}_i = (A - B_1B_1^TP_i(t) + \alpha^{-2}B_3B_3^TP_i(t))x_i(t)$.

Proof: Omitted here.

Remarks: (1) At some certain conditions, $t_i - t_{i-1}=$ constant, then $P_i(t)$ will be periodic. (2) All controllers still exist and can be constructed in the same way as above.

4 Conclusions

This paper presents some results about mixed disturbance attenuation using the H_∞ approach. These results can be considered a generalization of usual disturbance rejection by the H_∞ approach.

References

[1] E.B.Lee and L. Markus, Foundations of Optimal Control Theory, Wiley, N. Y.,1967.

[2] Q. F. Wei, W. P. Dayawansa, and P. S. Krishnaprasad, "H∞ Control for Impulsive Disturbances: A State-Space Solution," *Proc. ACC*, June, 1995, 4379-4383.

[3] J.C. Luo and E.B.Lee, "Mixed Disturbance Action for Stabilized Control Systems (Necessary Conditions)," *Proc. ACC*, June, 1999, 2563-2567.

[4] J.C. Luo and E.B. Lee, "Time-Optimal Control of the Swing Using Impulse Control Actions," *Proc. ACC*, June, 1998, 200-204.

[5] J.C. Luo and E.B. Lee, "Synthesis of Time-Optimal Feedback Controllers with Mixed Control Bang-bang and Impulse Actions," *Proc. Thirty-Sixth Annual Allerton Conference*, Sept., 1998, 470-478.

[6] Tamer Basar and Pierre Bernhard, "H^∞ -Optimal Control and Related Minimax Design Problems; A Dynamic Game Approach", Second Edition, Birkhauser, Boston 1995.

[7] Wayne W. Lu, Gary J. Balas and E.B.Lee, "Linear Quadratic Performance with Worst Case Disturbance Rejection" *Proc. of ACC*, (1995) 1962-1966.

[8] Mark M. Kogan, "Solution to the Inverse Problem of Minimax Control and Worst Case Disturbance for Linear Continuous-Time Systems" *IEEE Trans. on A.C.*, vol. 43, No. 5, 670-674, may 1998.

[9] Michael Green and David J. N. Limebeer, Linear Robust Control, Prentice-hall Inc., N. J., 1995.

Solving Polynomial Systems in Robust Stability Analysis

Nainn-Ping Ke
Department of Electrical Engineering - Systems
University of Southern California
Los Angeles, CA 90089-2563, U.S.A.
email: nke@eudoxus.usc.edu

Abstract

The objective of this paper is to describe a new method in robust stability analysis which the stability margin (k_M) problem can be formulated as solving polynomial systems by using stratified singularity theory and symbolic computation. Once the solutions are found, the stability margin problem can be easily solved. However, for real k_M problem, there could be thousands or even millions polynomial systems which need to be solved in order to find all singularities to determine whether the boundary of Horowitz template intercept the origin or not. In addition, there could be thousands or even millions edges which are critical. These reasons make singularity related method hard to have an efficient algorithm to compute real k_M. Fortunately, complex k_M problem is different from real case, no matter how many uncertainties, only one polynomial systems needs to be solved for the same purpose. In addition, there is no critical edges for complex k_M computation. Therefore, we might expect there will be an efficient algorithm of singularity related method to compute complex k_M.

1 Introduction

Robust control refers to the control of a plant with significant uncertainty by using a fixed controller and is a main branch of modern control theory [1, 2, 3]. Stability robustness analysis is a basic and fundamental topic in robust control. Robustness margin, k_M, is a measure of the stability analysis of a feedback system. However, it is a known result that robustness margin problem is NP-hard. Thus, pursuing the exact calculation of stability robustness margin is a tough job for control engineers. Grid method is a popular way to compute k_M. However, the dimension of the uncertainty space in robust stability analysis is normally high, for example, 20. If we choose 20 sampling points in each uncertainty, it means that 20^{20} evaluations must be made. This task is beyond the ability of any current supercomputer. Therefore, the motivation of this research is to find an alternative way to compute k_M without using the grid method and it has the same accuracy.

Safonov and Athans introduced multivariable stability margin (k_M) [8]. On the other hand, a framework for dealing with structured uncertainties is the "structured singular value" μ which was introduced by Doyle [9]. μ is the reciprocal of k_M. The standard method is to use the convex hull of the template to compute k_M instead of the template.

On the contrary, our method is to use the boundary of the template to compute k_M, not the convex hull [8, 14, 16, 18].

Singularity theory has been developed by Morse, Whitney, Mather, Thom, Milnor, Zeeman, Smale, Arnold and many other researchers for the decades [12, 14, 15, 17]. Kraus, Anderson and Mansour were first to report the use of the singularity for stability analysis [20]. A method based on Stratified Morse theory to compute real k_M by Jonckheere and Ke in [13]. Based on Symbolic Computation, this method has been implemented by Ke in [16]. However, due to this method dealing with real k_M, thousands or even millions polynomial systems need to be solved in order to obtain the singularities (See Table 1.). In addition, the singularity set includes thousands or even millions edges. There are the reasons why this method was not attractive at first. Fortunately, for complex k_M computation, different from real case, there is only one polynomial system which need to be solved for the same purpose [18]. In addition, there is no critical edges for complex k_M computation. Moreover, Computer Algebra (Symbolic Computation) Systems are more mature and more widely used in control engineer [10, 21]. By using Computer Algebra Systems, complex k_M problem can be easily formulated as solving ploynomial system.

A standard stability margin problem is shown in Figure 1. Let $f : \mathbb{D} \times \Omega \ni (q, \omega) \longmapsto det(I + L(j\omega)\Delta) \in \mathbb{C}$ be Nyquist Map, where $\mathbb{D}$ is the space of uncertain parameters (e.g., the Kharitonov cube or a polydisc) and Ω is frequency spectrum. $L(j\omega) \in \mathbb{C}^{n \times n}$ is the stable loop transfer matrix. q is the vector of parametric uncertainty which can be real or complex. The Horowitz template is $N_\omega = f(\mathbb{D}, \omega)$ [12, 14].

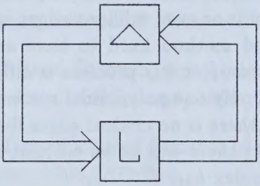


Figure 1: A standard robust stability problem

The maximum uncertainty bound, k_M, such that the system is stable for all $q \in \mathbb{D}$ with $\|q\|_\infty \leq k_M$ is called the robustness margin of a control system. In the above $\|q\|_\infty \overset{\text{def}}{=} \max_i\{\mid q_i \mid\}$ where $\mid q_i \mid$ denotes the absolute value of q_i for real uncertainties and the modulus of q_i for complex uncertainties. If we fix frequency, the k_M problem becomes to find the smallest solutions, $\|\Delta\|_\infty$, of $det(I + L\Delta) = 0$ which is a polynomial equation of unknowns. A critical point q^* of a smooth map f_ω defined over a smooth manifold is a point where the rank of the Jacobian drops [12, 16]. Therefore, the number of real constraints for a smooth critical point in the case of n real uncertainties is equal to the number of the real variables minus one (n-1). The boundary of the Horowitz template is contained in the image of the singularity set.

2 k_M Computation

Uncertainty space in k_M problem is non-manifold. Due to the difficulty in dealing with the non-manifold, the concept of stratified spaces is preferred.

- The idea for searching the singularities of the real uncertainty space is as follows: Real uncertainty space for 3-uncertainties is similar to a Kharitonov cube [12, 17]. Although the Kharitonov cube is not a manifold, it can be composed into manifolds, i.e.,

  closed cube = open cube ∪ { open faces } ∪ {open edges} ∪ {vertices}.

  Hence, we can apply the Morse theory to each manifold. The situation can be more complicated when deal with higher dimensional cases. (see Table 1.)

- The idea for searching the singularity of the complex uncertainty space is as follows: Complex uncertainty space for 2-uncertainties is similar to a solid doughnut which is a manifold with boundary. It can be decomposed into two smooth manifolds, i.e.,

  $$\text{solid doughnut} \;=\; \text{open doughnut} \;\cup\; \text{torus} \;.$$

  Hence, the Morse theory can be applied on both manifolds for singularities. In addition, due to lemma 1 of 1982 Doyle's paper [9] or Open Mapping Theorem [14], the image of the distinguished boundary is first to intercept the origin not the open doughnut. Therefore, only the singularities on the distinguished boundary, i.e., $\mathbb{T}^2$, are useful for k_M computation. We can used the **polar form** to represent the boundary of any size of polydisc with $z_i = ke^{j\phi_i}$, where k is the size of the polydisc. For example, in case of 2 complex uncertainties; the distinguished boundary of the polydisc, $S^1 \times S^1$, is a 2-dimensional manifold which can be expressed as $f_\omega^\phi = f_\omega(\phi_1, \phi_2)$. The Jacobian becomes $J^h_{f_\omega^\phi} = \begin{pmatrix} \frac{\partial \Re f_\omega^\phi}{\partial \phi_1} & \frac{\partial \Re f_\omega^\phi}{\partial \phi_2} \\ \frac{\partial \Im f_\omega^\phi}{\partial \phi_1} & \frac{\partial \Im f_\omega^\phi}{\partial \phi_2} \end{pmatrix}$. The constraint for the singularities is $\frac{\partial \Re f_\omega^\phi}{\partial \phi_1}\frac{\partial \Im f_\omega^\phi}{\partial \phi_2} - \frac{\partial \Im f_\omega^\phi}{\partial \phi_1}\frac{\partial \Re f_\omega^\phi}{\partial \phi_2} = 0$. It is easy to extend this procedure to general case. Thus, no matter how many uncertainties, there is only one polynomial system need to be solved. (see [18] for details.)

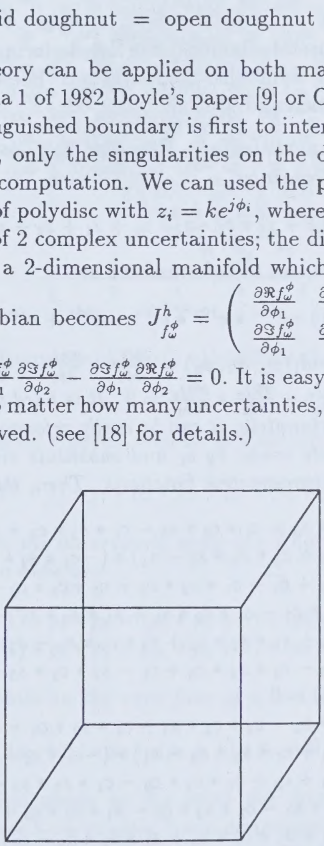


Figure 2: A Kharitonov cube.

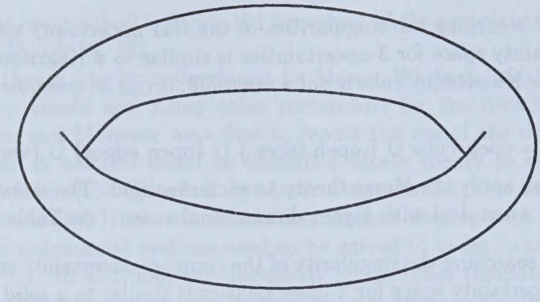


Figure 3: A torus.

3 Examples

The following examples are used to demonstrate how to formulate k_M problems as solving polynomial systems by using Symbolic Computation. These examples has been implemented in Maple, Mathematica and Matlab.

Example 1 (Complex Uncertainties) *Consider the following Nyquist map with complex uncertainties for fixed frequency*

$$f_\omega = (1+j) * z_1 * z_2 * z_3 + z_1 + z_2 + z_3 + 4 \tag{1}$$

where $\mid z_i \mid \leq 1$. *This problem can be reformulated as*

$$f_\omega^\phi = (1+j) * e^{j\phi_1} * e^{j\phi_2} * e^{j\phi_3} + e^{j\phi_1} + e^{j\phi_2} + e^{j\phi_3} + 4$$

Then, the constraints are $cond_1(\phi_1, \phi_2, \phi_3) = \frac{\partial \Re f_\omega}{\partial \phi_1} * \frac{\partial \Im f_\omega}{\partial \phi_2} - \frac{\partial \Re f_\omega}{\partial \phi_2} * \frac{\partial \Im f_\omega}{\partial \phi_1} = 0$ *and* $cond_2(\phi_1, \phi_2, \phi_3) = \frac{\partial \Re f_\omega}{\partial \phi_1} * \frac{\partial \Im f_\omega}{\partial \phi_3} - \frac{\partial \Re f_\omega}{\partial \phi_3} * \frac{\partial \Im f_\omega}{\partial \phi_1} = 0$. *It is hard to solve this set of nonlinear trigonometric equations. Fortunately, it can be easily reformulated as a set of polynomial equations as follow: substitute* $cos\phi_1$ *by* c_1 *and substitute* $sin\phi_1$ *by* s_1. *Use the similar substitutions for the other trigonometric functions. Then, the constraints become*

$$\left.\begin{array}{l}
(-s_1 * c_2 * c_3 + s_1 * s_2 * s_3 - c_1 * c_2 * s_3 - c_1 * s_2 * c_3 + s_1 * c_2 * s_3 + \\
s_1 * s_2 * c_3 - c_1 * c_2 * c_3 + c_1 * s_2 * s_3 - s_1) * (-c_1 * s_2 * c_3 - c_1 * c_2 * s_3 + \\
s_1 * s_2 * s_3 - s_1 * c_2 * c_3 + c_2 - c_1 * s_2 * s_3 + c_1 * c_2 * c_3 - s_1 * s_2 * c_3 - \\
s_1 * c_2 * s_3) - (-c_1 * s_2 * c_3 - c_1 * c_2 * s_3 + s_1 * s_2 * s_3 - s_1 * c_2 * c_3 + \\
c_1 * s_2 * s_3 - c_1 * c_2 * c_3 + s_1 * s_2 * c_3 + s_1 * c_2 * s_3 - s_2) * (-s_1 * c_2 * c_3 + \\
s_1 * s_2 * s_3 - c_1 * c_2 * s_3 - c_1 * s_2 * c_3 + c_1 - s_1 * c_2 * s_3 - s_1 * s_2 * c_3 + \\
c_1 * c_2 * c_3 - c_1 * s_2 * s_3) = 0 \\
(-s_1 * c_2 * c_3 + s_1 * s_2 * s_3 - c_1 * c_2 * s_3 - c_1 * s_2 * c_3 + s_1 * c_2 * s_3 + \\
s_1 * s_2 * c_3 - c_1 * c_2 * c_3 + c_1 * s_2 * s_3 - s_1) * (-c_1 * c_2 * s_3 - c_1 * s_2 * c_3 - \\
s_1 * c_2 * c_3 + s_1 * s_2 * s_3 + c_3 + c_1 * c_2 * c_3 - c_1 * s_2 * s_3 - s_1 * c_2 * s_3 - \\
s_1 * s_2 * c_3) - (-c_1 * c_2 * s_3 - c_1 * s_2 * c_3 - s_1 * c_2 * c_3 + s_1 * s_2 * s_3 - \\
c_1 * c_2 * c_3 + c_1 * s_2 * s_3 + s_1 * c_2 * s_3 + s_1 * s_2 * c_3 - s_3) * (-s_1 * c_2 * c_3 + \\
s_1 * s_2 * s_3 - c_1 * c_2 * c_3 - c_1 * s_2 * c_3 + c_1 - s_1 * c_2 * s_3 - s_1 * s_2 * c_3 + \\
c_1 * c_2 * c_3 - c_1 * s_2 * s_3) = 0 \\
s_1^2 + c_1^2 = 1 \\
s_2^2 + c_2^2 = 1 \\
s_3^2 + c_3^2 = 1
\end{array}\right\} \tag{2}$$

Hence, we have to solve this one-dimensional polynomial system to find the singularities. The simulation results for the template and the boundary are shown in Figure 4 and Figure 5, respectively.

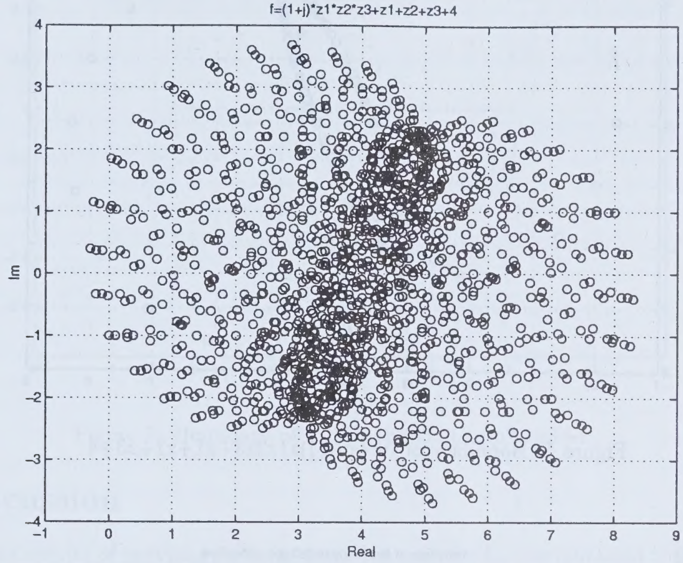


Figure 4: Template of fw=(1+j)z1z2z3+z1+z2+z3+4

Example 2 *Consider the same Nyquist map as in Example 1 with real uncertainties*

$$f_\omega = (1 + j) * p_1 * p_2 * p_3 + p_1 + p_2 + p_3 + 4$$

where $| p_i | \leq 1$. *For real case, the uncertainty space is a Kharitonov cube. The constraints of critical points on the open cube are*

$$\left.\begin{array}{l}(p_2 * p_3 + 1) * p_1 * p_3 - (p_1 * p_3 + 1) * (p_2 * p_3) = 0 \\ (p_2 * p_3 + 1) * p_1 * p_2 - (p_1 * p_2 + 1) * (p_2 * p_3) = 0\end{array}\right\} \tag{3}$$

The constraint of critical points on the open face of $p_1 = 1$ *is*

$$(p_3 + 1) * p_2 - (p_2 + 1) * p_3 = 0 \tag{4}$$

The constraints of the critical points on the other five faces can be obtained similarly. Therefore, for this problem, we have to solve 7 one-dimensional polynomial systems to find the singularities. In addition, the singularity set also includes 12 edges. The simulation results for the template and the boundary are shown in Figure 6 and Figure 7, respectively.

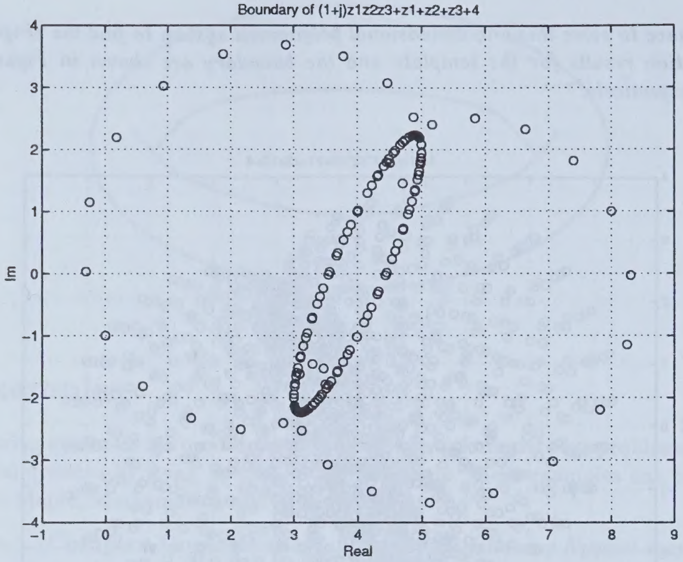


Figure 5: Boundary of fw=(1+j)z1z2z3+z1+z2+z3+4

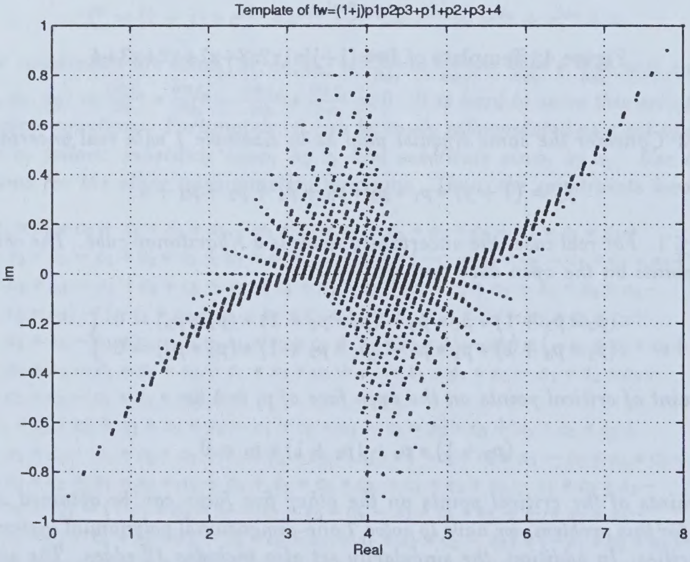


Figure 6: f=(1+j)p1p2p3+p1+p2+p3+4

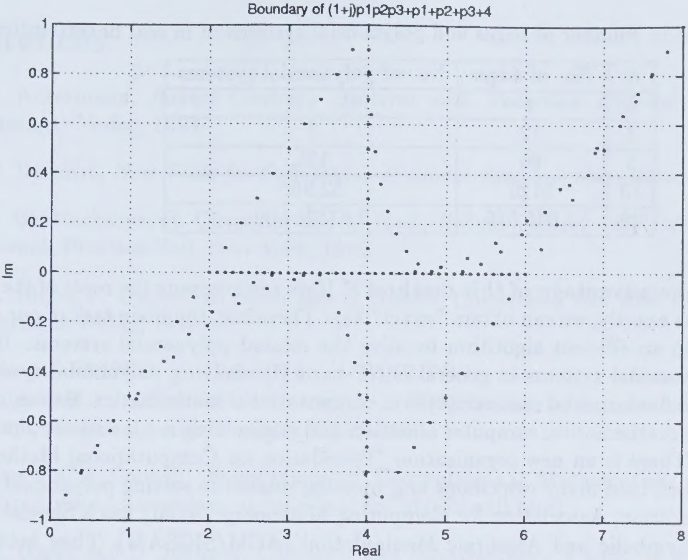


Figure 7: Boundary of (1+j)p1p2p3+p1+p2+p3+4

4 Discussion

Based on the results of previous sections, the main task for k_M computation is to solving polynomial systems. Hence, we could ask the following questions ?

"**Is complex k_M problem easier than real k_M problem ?**". The answer is not quitely true due to the complex robustness margin problem has higher total degree than real one (compare Example 1 and 2). We need to solve thousands or even millions sets of polynomial equations to determine the boundary of the template for real uncertainties (See Table 1) [1, 2]. On the contrary, for the complex case, even with higher dimensional cases, "We only need to solve one polynomial system to determine the boundary ." makes complex k_M problem more likely to be solved in a short time than the real one. Moreover, the reasons for us to expect that there will be a good algorithm to compute complex k_M are the followings:

1. Only one polynomial system needs to be solved for complex k_M.
2. The solutions which are real, and Morse critical points, and lie between -1 and 1 are useful.

Based on the earlier discussion, it is hard to implement our approach to compute k_M under the pure numeric computing environment. The reasons are the followings:

1. It needs to partially differentiate the Nyquist map relative to q_i or ϕ_i.
2. It needs to change and/or substitute variables.
3. It needs to solve polynomial systems.

Table 1: Number of edges and polynomial systems of m real uncertainties

m	No. of edges	No. of polynomial systems
2	4	1
3	12	7
5	80	131
10	5120	52,905
20	1,048,576	3.4753e+09

"**What is the advantage of this method** ?" If we can compute the roots of the polynomial systems exactly, we can obtain "exact" k_M. Therefore, the main task of our research is to develop an efficient algorithm to solve the related polynomial systems. However, solving polynomial systems in general is NP-hard [7]. Solving polynomial systems is a classical and fundamental research topic in computational mathematics. Hence, many researchers in mathematics, computer scientists and engineering are facing the similar task as we are. There is an new organization "Foundation on Computational Mathematics" (FoCM) which held many workshops and meeting related to solving polynomial systems [4, 6]. In addition, Association for Computing Machinery (ACM) has a Special Interest Group on Symbolic and Algebraic Manipulation (ACM/SIGSAM). They hold an annual International Symposium on Symbolic and Algebraic Computation(ACM/SIGSAM ISSAC) which is closedly related to our research [10, 11]. Moreover, there is a three-year project, FRISCO, funded by European Commission [5]. The aim of the FRISCO project is to investigate and develop technologies which can be used to deliver highly, efficient, versatile polynomial solvers for industrial use. In addition, the predecessor PoSSo project developed algebraic library for solving polynomial systems. Furthermore, many Computer Algebra Systems such as Maple, Mathematica, AXIOM, Reduce, etc., provide many symbolic/numeric functions to solve polynomial systems [10, 11]. Hopefully, in the near future, we might find a good algorithm to compute complex k_m.

5 Concluding Remarks

It has shown that the robustness margin problem can be formulated as solving polynomial systems by Symbolic Computation. The main task of this approach is to find the solutions which are the Morse critical points. However, due to the high dimension of the uncertainty space, for example, 20 uncertainties, which could be a real world problem, the relating polynomial system is huge, for example, 39 polynomial equations for 20 complex uncertainties. Hence, "Can we solve these kind of huge polynomial systems ?" becomes a main issue of our future research.

Acknowledgement

I am grateful to Prof. E. A. Jonckheere and Prof. M. G. Safonov, Department of Electrical Engineering - Systems, University of Southern California, for many helpful suggestions. I also like to thank Prof. J. V. Medanic, Department of General Engineering, University of Illinois at Urbana-Champaign, for his encouragement.

References

[1] J. Ackermann, *Robust Control - Systems with Uncertain Physical Parameters*, Springer-Verlag, 1993.

[2] B. Barmish, *New Tools for Robustness of Linear Systems*, Macmillan, 1994.

[3] S. Bhattacharya, H. Chapellat and L. Keel, *Robust Control - the Parametric Approach*, Prentice-Hall, New York, 1995.

[4] L. Blum, F. Cucker, B. Shub and S. Smale, *Complexity and Real Computation*, Springer-Verlag, 1997.

[5] The FRISCO Consortium, *FRISCO - A Framework for Integrated Symbolic/Numeric Computation*, available at http://www.nag.co.uk/Projects/Frisco.html.

[6] The FoCM Council, Foundations of Computational Mathematics, available at http://www/dampt.cam.uk/user/na/FoCM/Council.html.

[7] M. Garey and D. Johnson, *Computer and Intractability - A guide to the Theory of NP-completeness*, W. H. Freeman, 1979.

[8] R. de Gaston and M. Safonov, *Exact Calculation of the Multiloop Stability Margin*, IEEE Trans. on Automat. Contr., vol. 33, pp. 156–171, 1988.

[9] J. Doyle, *Analysis of Feedback System with Structured Uncertainty*, IEE Proc., vol. 129, pt. D. no. 6, pp. 242–250, 1982.

[10] K. O. Geddes, S. R. Czapor and G. Labahn, *Algorithms for Computer Algebra*, Kluwer Academic Publishers, 1992.

[11] J. von zur Gathen and J. Gerhard, Modern Computer Algebra, Cambridge University Press, 1999.

[12] E. A. Jonckheere, *Algebraic and Differential Topology of Robust Stability*, Oxford University Press, Oxford, New York, 1997.

[13] E. A. Jonckheere and N. P. Ke, Stability Margin relative to Stratified Uncertainty Space, Proc. IEEE Conf. on Decision and Control (CDC), pp. 1322–1323, New Orleans, Louisiana, 1995.,

[14] E. A. Jonckheere and N. P. Ke, *Complex-Analytic Theory of the μ-Function*, Proc. American Control Control (ACC), pp.366-371, Albuquerque, New Mexico, 1997; also to appear J. Math. Anal. Appl., 1999.

[15] E. A. Jonckheere and N. P. Ke, Real versus Complex Robustness Margin Continuity as a Smooth versus Holomorphic Singularity Problem, Proc. IEEE CDC, pp. 3266–3271, San Diego, 1997; also to appear J. Math. Anal. Appl., 1999.

[16] N. P. Ke, *An Algorithm of Symbolic Computation for Stability Margin*, Proc. ACC, pp. 2304–2308, Philadelphia, Pennsylvania, 1998.

[17] N. P. Ke, *Stratified Morse Theory for Stability Margin Computation*, Proc. 36th Annual Allerton Conf. on Communication, Control and Computing, pp. 489–498, Monticello, Illinois, 1998.

[18] N. P. Ke, *A New Method to Compute Complex Robustness Margin*, Proc. IFAC'99 World Congress, vol. G, pp. 37–42, Beijing, China, 1999.

[19] N. P. Ke, *Symbolic and Algebraic Computation in Robust Stability Analysis*, Booklet of Posters Abstract, pp. 27–30, ACM/SIGSAM International Symposium on Symbolic and Algebraic Computation, Vancouver, Canada, July 28–31, 1999.

[20] F. J. Kraus, B. O. D. Anderson and M. Mansour, *Robust Stability of Polynomials with Multilinear Parameter Dependence*, Int. J. Control, pp. 1745–1762, 1989.

[21] N. Munro (ed.), *Symbolic Methods in Control System Analysis and Design*, IEE Publications, 1999.

Robust, Near Time-Optimal Control of Third-Order Uncertain Systems

K. H. You and E. B. Lee
University of Minnesota, ECE Dept., 200 Union Street, Minneapolis, MN 55455
ykh@ece.umn.edu and eblee@ece.umn.edu

1 Introduction

For real systems, much effort has been devoted to minimize the time-optimal performance index. Time-optimal control schemes usually involve bang-bang control selected by a state feedback(switch curve) [1]. However it is not good in the sense of system robustness. With some uncertainties(parameter uncertainty, modeling error, disturbance), the controller oscillates rapidly near the goal state during the finite time response. This chattering effect of the controller can make the system unstable.

Because of these undesirable characteristics, several modified schemes have been proposed which are more robust and nearly time-optimal. Pao proposed a proximate time-optimal control of third-order servomechanism($PTOS$) [2]. Unfortunately there is a limitation by the slight change of system. Adding a pole or zero to system equations makes it hard to compute the time optimal-controller. For nonlinear higher-order systems, mathematical tools have been developed with the use of sliding modes in the so-called *Variable Structure System*(VSS) [3]. The nonlinear state feedback controller makes the state change according to the switching logic.

2 Time-Optimal Controller Design Principles

2.1 Type-2 third-order system

Consider the linear third-order system given by state model

$$\dot{\mathrm{x}} = A\mathrm{x} + Bu + D \tag{1}$$

where

$$A = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & -a+\delta(t) \end{bmatrix}, \ B = \begin{bmatrix} 0 \\ 0 \\ b \end{bmatrix}, \ D = \begin{bmatrix} 0 \\ 0 \\ d(t) \end{bmatrix}.$$

$\mathrm{x}^T \triangleq [x_1 \ x_2 \ x_3]$ is the system state, a is real positive parameter, while $\delta(t)$ is the parameter uncertainty of the system. $d(t)$ is an external disturbance and u is the scalar control function. It is also assumed that the admissible control values lie in a closed and bounded region. Especially the upper bounds of $\delta(t)$ and $d(t)$ are supposed to be known as $|\delta(t)| \le \bar{\delta}$ and $|d(t)| \le \bar{d}$.

For time-optimal control, we consider the performance index as $J_1 = \frac{1}{2}\int_0^{t_f} dt$, ($t_f$ is not determined). Now we have the transformed system with the state variables$(y_1(t), y_2(t), y_3(t))$.

$$\dot{y} = \hat{A}y + \hat{B}u + \hat{D} \tag{2}$$

where

$$\hat{A} = \begin{bmatrix} 0 & a & -\delta(t) \\ 0 & 0 & \delta(t) \\ 0 & 0 & -a+\delta(t) \end{bmatrix}, \ \hat{B} = \begin{bmatrix} -a \\ a \\ a \end{bmatrix}, \ \hat{D} = \frac{a}{b}\begin{bmatrix} -d(t) \\ d(t) \\ d(t) \end{bmatrix}.$$

In this paper, we take the desired target set as origin for convenience. The time-optimal control for the system given by Eqs.(1)-(2) without parameter uncertainty and external disturbance($\delta(t) = d(t) = 0$) is as follows;

caseI : when the state, (y_1, y_2, y_3) lies above or below the switch surface, S_f,

$$u_1^* = -U_{sat}sgn\{f_s(y_1, y_2, y_3)\} \tag{3}$$

caseII : when the state, (y_1, y_2, y_3) lies on the switch surface, S_f,

$$u_1^* = -U_{sat}sgn\{f_c(y_1, y_2)\} \tag{4}$$

caseIII : when the state, (y_1, y_2, y_3) lies on the switch curve, S_c

$$u_1^* = -U_{sat}sgn(y_2) \tag{5}$$

where

$$\begin{aligned} f_s(y_1, y_2, y_3) &= y_3 - sgn(y_2) + sgn(y_2)e^{\{sgn(y_2)+[-sgn(y_2)f_{c1}(y_1,y_2)]^{\frac{1}{2}}\}} \\ &\quad \cdot(2 - e^{\{[-sgn(y_2)f_{c1}(y_1,y_2)]^{\frac{1}{2}}\}}) && (6) \\ f_c(y_1, y_2) &= y_1 + y_2 + \frac{1}{2}sgn(y_2)y_2^2 && (7) \end{aligned}$$

2.2 Sliding mode control

Implementation of bang-bang time-optimal control is impractical since it is very sensitive to modeling uncertainties. Especially with unpredicted disturbances and parameter uncertainties near the goal state, the control will change rapidly from one value to another on the switch surface or switch curve. This is called *chattering.* The chattering can happen in most real systems. But this can be reduced using *sliding mode control.* It improves the stability and robustness with the sliding mode. This sliding mode gives the system some invariant properties which are independent of parameter uncertainties and disturbances. Sliding mode induces the state to move toward the sliding surface, $s(x_1, x_2, x_3) = 0$. To start with use the parametric surface

$$s(x_1, x_2, x_3) = x_3 + c_1x_2 + c_2x_1 \tag{8}$$

where c_1, c_2 are strictly positive constants. Let a Lyapunov function candidate be $\text{V} = \frac{1}{2}s^2$. Then along the solution curves on the surface,

$$\dot{\text{V}} = s[(-a + \delta(t))x_3 + bu + d(t) + c_1x_3 + c_2x_2].$$

To realize Lyapunov stability condition, choose the nonlinear feedback controller,

$$\begin{aligned} u_2^* &= \frac{1}{b}[(a - c_1)x_3 - c_2x_2 - \Upsilon sat(s)] && (9) \\ \text{with } \Upsilon &= \bar{\delta}|x_3| + \bar{d} + \eta, \quad \eta > 0. \end{aligned}$$

This ensures that the derivative is negative along the solution curves. To avoid the chattering effect, we used the saturation function, $sat(\cdot)$ as the smooth controller in Eq.(9): $sat(s) = 1 \ \ for\ s > 1,\ s \ \ for -1 \le s \le 1,\ -1 \ \ for\ s < -1.$

2.3 Time-variant sliding mode control:(*TV SMC*)

With the change of performance index around the origin, we can design a new sliding mode control in which the sliding surface depends on the initial values minimizing the chattering effect. Choose for this the index $J_2 = \int_T^\infty |e(t)|\,dt,$ when $\mathrm{x}Q\mathrm{x} \le k,\ \ k > 0.$ Here $e(t) = x_1(t) - x_r(t)$ and T is the time when the states move into an ellipsoid defined by $\mathrm{x}Q\mathrm{x} = k$. Now we are to find another sliding mode control with the performance index, J_2. In this region, let's take a different sliding surface, $s(x_1, x_2, x_3) = 0$ given by

$$\begin{aligned} s(x_1, x_2, x_3) &= x_3 + 2\alpha x_2 + \alpha^2 x_1 \\ &= \ddot{e} + 2\alpha\dot{e} + \alpha^2 e, \quad \alpha > 0. \end{aligned} \tag{10}$$

That is, we suppose the solution of error equation(Eq.(10))) has two multiple roots and decays to origin exponentially. And we assume that we have the initial values of e_1, e_2 at $t = T$, i.e, $e_1(T) = \xi_1, e_2(T) = \xi_2$ and $\xi_1\xi_2 < 0$.

$$e(t) = \xi_1 e^{-\alpha(t-T)} + (\xi_2 + \alpha\xi_1)(t - T)e^{-\alpha(t-T)}, \ \ t \ge T. \tag{11}$$

From the equation of index, J_2 and Eq.(11),

$$\begin{aligned} J_2 &= \int_T^\infty |e(t)|\,dt \\ &= \left|\frac{2\alpha\xi_1 + \xi_2}{\alpha^2}\right| \end{aligned} \tag{12}$$

and α has a finite value. To minimize this performance index, set

$$\alpha = -\frac{\xi_2}{2\xi_1}. \tag{13}$$

Following the procedure in the previous section,

$$\dot{\mathrm{V}} \ = \ s[(-a + \delta(t))x_3 + bu + d(t) + 2\alpha x_3 + \alpha^2 x_2].$$

Choose the nonlinear feedback controller,

$$\begin{aligned} u_3^* &= \frac{1}{b}[(a - 2\alpha)x_3 - \alpha^2 x_2 - \Upsilon sat(s)] \\ \Upsilon &= \bar{\delta}|x_3| + \bar{d} + \eta, \quad \eta > 0 \\ \alpha &= -\frac{\xi_2}{2\xi_1}. \end{aligned} \tag{14}$$

The proposed control algorithm invokes two different computations.

$$u^* = \begin{cases} u_1^* \text{ (bang-bang time-optimal control)}, & \text{when } \ \mathrm{y} = P^{-1}\mathrm{x},\ \mathrm{x}Q\mathrm{x} > k \\ \frac{1}{b}[(a - 2\alpha)x_3 - \alpha^2 x_2 - \Upsilon sat(s)],\ \ \alpha = -\frac{2\xi_2}{\xi_1}, & Otherwise. \end{cases} \tag{15}$$

3 Simulation results

Simulations have been run for time-optimal control and time-variant sliding mode control with the following parameters;$a = b = U_{sat} = 1; \bar{\delta} = \bar{d} = 0.1; k = 5; Q = 3 \times 3$ *Identity matrix*; $x_1(0) = -1; x_2(0) = 0; x_3(0) = -3;$ Therefore $\delta(t)$ can be thought as $0.1sin(t)$ and $d(t)$ as $0.1cos(2t)$. The problem is to find the optimal controller which steers any initial state to the final goal state, $x_r = 0$. We simulated dynamic behavior of the system controlled according to Eq.(15).

Simulation results show that the proposed control algorithm has nearly time-optimal move and robust to model uncertainty. Fig.1 shows that the trajectories of states from any initial values approach to the goal states in finite operation time and that the chattering of the controller near the origin is reduced compared with those in case of pure bang-bang control.

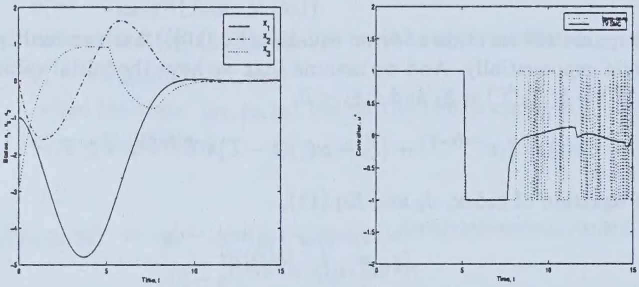


Figure 1: State trajectories and controller comparison

4 Conclusions

We have developed a robust, near time-optimal control algorithm for type-2 third-order linear system. The property of robustness in sliding mode control is used to overcome the shortcomings in pure bang-bang control. By analysis and simulations, it has been shown that the proposed control algorithm achieved system stabilization and approximate minimum settling time even with parameter uncertainty and disturbance.

References

[1] E.B. Lee and L. Markus, Foundations of Optimal Control Theory, Wiley, N. Y., 1967

[2] L.Y. Pao, "Proximate Time-Optimal Control of Thrid-Order Servomechanisms ," *IEEE Trans. Automat. Contr., 38* 1993, pp. 560-580

[3] J.Y. Hung, W. Gao and J.C. Hung, "Variable Structure Control: A Survey ," *IEEE Trans. Industrial Electronics, 40* 1993, pp. 2-22

A New Convex Relaxation for Robust H_2 Performance Analysis of Uncertain Linear Systems

Laurent El Ghaoui
University of California Berkeley
elghaoui@eecs.berkeley.edu

Eric Feron
M.I.T.
feron@mit.edu

abstract

This paper introduces a new formulation for the analysis of robust stability and H_2 performance of a linear system subject to parametric uncertainties. The problem is recast as a quadratically constrained, quadratic program. Convex, semidefinite relaxations yield conservative answers to this problem.

New Upper Bounds on Robust H_2 Performance

Consider the stable linear system

$$\frac{d}{dt}x = (A + B_p\delta C_q)x + B_w w, \quad z = C_z x, \quad x(0) = x_0, \tag{1}$$

where $\delta = \mathbf{diag}(\delta_1, \ldots, \delta_p)$ is a diagonal matrix with unkown coefficients with absolute value less than one. This paper is concerned with computing the system's worst-case H_2 performance under the stated conditions. In this paper, H_2 performance is given by $\lim_{t\to\infty} \mathbf{E}\, z^T z$ when w is white noise. The H_2 performance of the linear system (1) may be computed as $\mathbf{Tr}\, B_w^T P_{\mathrm{obs}} B_w$, where P_{obs} is the observability Gramian and satisfies

$$(A + B_p\delta C_q)^T P_{\mathrm{obs}} + P_{\mathrm{obs}}(A + B_p\delta C_q) + C_z^T C_z = 0. \tag{2}$$

For any symmetric matrix R, denote $\overline{R}$ its vector representation: If

$$R = \begin{bmatrix} r_{11} & \cdots & r_{1q} \\ \vdots & \ddots & \vdots \\ r_{m1} & \cdots & r_{mq} \end{bmatrix}, \text{ then } \overline{R}^T = [r_{11} \cdots r_{1q} \quad \cdots \quad r_{m1} \cdots r_{mq}].$$

Conversely, for a vector v with appropriate dimension, denote $\underline{v}$ its matrix representation. Define the linear operator $\mathcal{A}$ so that $\mathcal{A}\overline{P} = \overline{A^T P + PA}$. Then from (2) we have

$$\overline{P}_{\mathrm{obs}} = -\mathcal{A}^{-1}\overline{(C_z^T C_z + C_q^T \delta B_p^T P_{\mathrm{obs}} + P_{\mathrm{obs}} B_p \delta C_q)}$$

Introducing $Q_{\mathrm{obs}} = \delta B_p^T P_{\mathrm{obs}}$, the value of P_{obs} can be written as

$$\overline{P}_{\mathrm{obs}} = -\mathcal{A}^{-1}\overline{(C_z^T C_z + C_q^T Q_{\mathrm{obs}} + Q_{\mathrm{obs}}^T C_q)}.$$

so that computing the worst-case H_2 norm may be formulated as an optimization over Q_{obs}:

$$\begin{array}{ll} \text{maximize} & \mathcal{C}^T\overline{Q}_{\mathrm{obs}} + \mathcal{D} \\ \text{subject to} & \overline{Q}_{\mathrm{obs}} = \Delta(\mathcal{F}\overline{Q}_{\mathrm{obs}} + \mathcal{G}), \end{array} \quad \text{where} \quad \begin{array}{l} \mathcal{C}^T\overline{Q} + \mathcal{D} \triangleq -\mathbf{Tr}\, B_w^T \underline{\mathcal{A}^{-1}\overline{(C_z^T C_z + C_q^T Q + Q^T C_q)}} B_w \\ \mathcal{F}\overline{Q} + \mathcal{G} \triangleq -\overline{B_p^T \underline{\mathcal{A}^{-1}\overline{(C_z^T C_z + C_q^T Q + Q^T C_q)}}}, \end{array}$$

Thus $\mathcal{C}$, $\mathcal{D}$, $\mathcal{E}$ and $\mathcal{F}$ are uniquely defined matrices and $\Delta = \mathbf{diag}(\underbrace{\delta_1,\ldots,\delta_1}_{n},\ldots,\underbrace{\delta_p,\ldots,\delta_p}_{n})$.

This problem may be written as a nonconvex, quadratically constrained quadratic program in the variable Q_{obs} by noticing that

$$\overline{Q}_{\mathrm{obs}} = \Delta(\mathcal{F}\overline{Q}_{\mathrm{obs}} + \mathcal{G}) \Leftrightarrow \begin{bmatrix} \overline{Q}_{\mathrm{obs}} \\ \mathcal{F}\overline{Q}_{\mathrm{obs}} + \mathcal{G} \end{bmatrix}^T \begin{bmatrix} S & -T \\ T & -S \end{bmatrix} \begin{bmatrix} \overline{Q}_{\mathrm{obs}} \\ \mathcal{F}\overline{Q}_{\mathrm{obs}} + \mathcal{G} \end{bmatrix} \leq 0 \tag{3}$$

for any matrices $S = S^T$ and $T = -T^T$ satisfying

$$S = \mathbf{diag}(S_1,\ldots,S_p), \ \ T = \mathbf{diag}(T_1,\ldots,T_p), \ \ (S_i, T_i) \in \mathbf{R}^n, i = 1,\ldots,p. \tag{4}$$

The procedure of replacing the uncertainty Δ by the family of quadratic relations (4) is standard in the control and systems literature. Thus the optimization problem is to maximize $\mathcal{C}^T\overline{Q}_{\mathrm{obs}}$ subject to (3), where the constraint has to hold for all admissible values of S and T, and its value is the worst-case $\mathbf{H}_2$ performance of the system (1). This problem is a quadratically constrained, quadratic program in the variable Q_{obs}. Introduce the Lagrangian

$$\mathcal{L}(\overline{Q}, S, T) = \mathcal{C}^T\overline{Q} - \begin{bmatrix} \overline{Q} \\ \mathcal{F}\overline{Q} + \mathcal{G} \end{bmatrix}^T \begin{bmatrix} S & -T \\ T & -S \end{bmatrix} \begin{bmatrix} \overline{Q} \\ \mathcal{F}\overline{Q} + \mathcal{G} \end{bmatrix}.$$

Then an upper bound on robust $\mathbf{H}_2$ performance is given by $\min_{S,T} \max_Q \mathcal{L}(\overline{Q}, S, T)$. It is easily shown this optimization problem is a convex, semidefinite program in S and T [1]. The proposed procedure to compute upper bounds on worst-case $\mathbf{H}_2$ performance clearly uses established methodologies in the area of robust control, since it relies on a positive semidefinite relaxation of the problem. However it differs from previous approaches by eliminating the traditional intermediate usage of parameter-dependent Lyapunov functions [2] or stability multipliers [3]. It would be interesting to investigate the links between the proposed method and these approaches. All derivations have been performed using the expression of $\mathbf{H}_2$ performance via the observability Gramian. It is clear that this approach could also be developed via the expression of the $\mathbf{H}_2$ norm of the system in terms of its controllability Gramian and could possibly lead to different numerical results.

References

[1] Boyd S. , El Ghaoui L. , Feron E. , and Balakrishnan V. (1994) *Linear Matrix Inequalities in System and Control Theory*, volume **15** of *SIAM Studies in Applied Mathematics* SIAM.

[2] Haddad W. and Bernstein D. Parameter-Dependent Lyapunov Functions, Constant Real Parameter Uncertainty, and the Popov Criterion in Robust Analysis and Synthesis (1991) In *Proc. IEEE Conf. on Decision and Control*, pages 2274–2279, 2632–2633, Brighton, U.K.

[3] Feron E. (1997) Analysis of robust $\mathbf{H}_2$ performance using multiplier theory. *SIAM J. on Control and Optimization.*

A new result on the Bellman equation for exit time control problems with critical growth dynamics

Michael Malisoff [1]
Rutgers University Mathematics Department
Piscataway, NJ 08854-8019, USA
malisoff@math.rutgers.edu

§1. Introduction

This note proves that the value function of Sussmann's Reflected Brachystochrone Problem (RBP) for an arbitrary singleton target $\{B\}$ is the unique viscosity solution of the corresponding Bellman equation on $\mathbb{R}^2 \setminus \{B\}$ among functions which vanish at B, are continuous in the plane, and are bounded below. The RBP is the time-optimal problem of bringing points in the plane to B using the dynamics $\dot{x} = u_1\sqrt{|y|}$, $\dot{y} = u_2\sqrt{|y|}$, subject to $(u_1, u_2) \in C := \{z \in \mathbb{R}^2 : ||z||_2 \leq 1\}$. For an analogous uniqueness characterization for a *general* class of exit time problems whose dynamics do not admit unique trajectories for some choices of inputs and initial positions, see [2]. We denote the value function of the RBP for the target $\{B\}$ by T_B (so $T_B(x)$ is the infimum of the times t for which some RBP trajectory brings x to B in time t). The corresponding Bellman equation is

$$|Dv((x,y))|\sqrt{|y|} - 1 = 0. \tag{1}$$

(If $F : \mathbb{R}^N \times \mathbb{R}^N \to \mathbb{R}$ is continuous, then a *viscosity subsolution* (resp., *supersolution*) of $F(q, Dv(q)) = 0$ on an open subset $\Omega \subset \mathbb{R}^N$ is a continuous function $w : \Omega \to \mathbb{R}$ so that $F(p, D\gamma(p)) \leq$ (resp., $\geq$) 0 for all C^1 functions $\gamma : \Omega \to \mathbb{R}$ and all local maxima (resp., minima) p of $w - \gamma$. A *viscosity solution* of $F(q, Dv(q)) = 0$ on Ω is then a function which is both a viscosity sub- and supersolution of that equation on Ω.)

§2. Proof that T_B is a viscosity solution of (1) on $\mathbb{R}^2 \setminus \{B\}$

Consider the problems obtained from the RBP by replacing the RBP dynamics with $\dot{x} = u_1\sqrt[4]{y^2 + 1/n}$, $\dot{y} = u_2\sqrt[4]{y^2 + 1/n}$ for each $n \in \mathbb{N}$ (with the same constraint on (u_1, u_2)). Call the n-th such problem P_n. By standard results from [1], we know that the value function $T_{n,B}$ for P_n is a viscosity solution of the corresponding Bellman equation

$$|Dv((x,y))|\sqrt[4]{y^2 + 1/n} - 1 = 0 \tag{2}$$

on $\mathbb{R}^2 \setminus \{B\}$. Standard stability results would then let us conclude that T_B is a viscosity solution of (1) on $\mathbb{R}^2 \setminus \{B\}$ if we show $T_{n,B} \to T_B$ unformly on compacta on $\mathbb{R}^2 \setminus \{B\}$. We prove this convergence by the Ascoli-Arzelá Theorem. First note that $T_{n,B} \leq T_B$ pointwise (since if (ϕ_1, ϕ_2) is a trajectory for the RBP for the input u, then it is also a trajectory for the P_n dynamics for $u|\phi_2|^{1/2}/(\phi_2^2 + 1/n)^{1/4}$). The definition of the infimum gives $|T_{n,B}(x) - T_{n,B}(y)| \leq T_y(x) \vee T_x(y)$ for $x, y \in \mathbb{R}^2$, and an elementary consideration of vertical and horizontal movements along RBP trajectories which we omit establishes that $(p,q) \mapsto T_p(q)$ is continuous. Fixing $p \in \mathbb{R}^2 \setminus \{B\}$, we can therefore let v_p denote a uniform limit of a subsequence of the $T_{n,B}$'s (which we do not relabel) on $B_\delta(p)$ for some $\delta > 0$ (depending on p). Again using stability, v_p is a viscosity solution of (1) on $B_\delta(p)$, so the result follows if we show $T_{n,B} \to T_B$ pointwise on $B_\delta(p)$. Fixing $\varepsilon \in (0,1)$ and $x \in B_\delta(p)$, this amounts to showing that $T_{n,B}(x) + \varepsilon > T_B(x)$ for n large enough. In what follows, $f_n, f : \mathbb{R}^2 \times C \to \mathbb{R}^2$ denote the dynamics for P_n and the RBP, respectively.

---

[1] This research was supported in part by NSF Grant DMS95-00798. It was carried out during January 1999, while the author was a Research Assistant for Héctor Sussmann. The author thanks Dr. Sussmann for his helpful comments on early drafts of this paper.

For each $n \in \mathbb{N}$, there is an input α_n so that the corresponding f_n trajectory $\phi_n = (\phi_{n,1}, \phi_{n,2})$ drives x to B in time $t_n \leq T_{n,B}(x) + \varepsilon/2$. Note that ϕ_n is also a trajectory for f_1 (using the input $\beta_n := \alpha_n(\phi_{n,2}^2 + 1/n)^{1/4}/(\phi_{n,2}^2 + 1)^{1/4}$). Now apply the classical compactness theorem for relaxed controls (cf. [1] and [4]). Passing to a weak-$\star$, Radon measure valued (subsequential) limit $\bar{\beta}$ of the β_n's, we obtain a relaxed trajectory ϕ^r for f_1 (meaning $\phi^{r\prime}(t) = \int_C f_1(\phi^r(t), a) d(\bar{\beta}(t))(a)$ for all t) so that $\phi_n \to \phi^r$ uniformly on compacts (along a subsequence). Assuming wlog that $\alpha_n \to \bar{\alpha}$ weak-$\star$, it follows that

$$\phi^r(t) \leftarrow \phi_n(t) = x + \int_0^t f_n(\phi_n(s), \alpha_n(s))\, ds \to x + \int_0^t \int_C f(\phi^r(s), a) d(\bar{\alpha}(s))(a)\, ds$$

for all $t \geq 0$. Since f is affine in the input, it follows that there is an input α so that (α, ϕ^r) is an input-trajectory pair of f. Since $t_n \leq T_B(x) + 1$ for all n, we assume $t_n \to \mu \in \mathbb{R}$. It follows that $\phi^r(\mu) = B$, so $T_{n,B}(x) + \varepsilon \geq T_B(x)$ for large n, as desired.

§3. Proof of uniqueness characterization

Let $w \in C(\mathbb{R}^2)$ be a viscosity solution of (1) on $\mathbb{R}^2 \setminus \{B\}$ which is bounded below and vanishes at B. By standard comparison theorems (e.g., Theorem IV.4.3 of [1]), we get $w \geq T_{n,B}$ pointwise for all n, since w is also a viscosity supersolution of (2) on $\mathbb{R}^2 \setminus \{B\}$, so $w \geq T_B$ pointwise, since $T_{n,B} \to T_B$ pointwise. These theorems apply since the f_n's are *uniformly Lipschitz*, i.e., Lipschitz in the state variable uniformly in the control value.

To prove the reverse inequality, first note that any RBP trajectory ψ starting at a $p \in \mathbb{R}^2$ that first reaches B at time $s > 0$ admits $\mu_1, \mu_2 \in (0, s)$ with $\mu_2 \leq \mu_1$ and an RBP subtrajectory $\tilde{\psi}$ on $[0, \mu_2]$ with $\tilde{\psi}(\mu_2) = \psi(\mu_1)$ such that $\tilde{\psi}\lceil[1/n, \mu_2] \cap \{y = 0\} = \emptyset$ for large n. To see why, assume wlog that $p = (p_1, 0)$. Note that f does not allow movement along the x-axis. It follows that $\psi(\tilde{t})$ lies in $\{y > 0\}$ or $\{y < 0\}$ at some $\tilde{t} \in (0, s)$. Assume $\psi(\tilde{t}) \in \{y > 0\}$ wlog. Reflect the subtrajectories of ψ which lie in $\{y < 0\}$ over the x-axis to get another RBP trajectory $\hat{\psi}$ that first reaches $\psi(\tilde{t})$ at some time $\hat{t} \in (0, \tilde{t}]$ and lies in the closed upper half plane. From [3], one optimal RBP trajectory joining points P and Q in the closed upper half plane is a cycloid arc passing from P to Q without hitting the x-axis in between. Replace $\hat{\psi}\lceil[0, \hat{t}]$ with such an arc $\tilde{\psi}$ for $P = p$ and $Q = \hat{\psi}(\hat{t})$ which first reaches $\psi(\tilde{t})$ at μ_2 and set $\mu_1 = s$.

Fix $x \in \mathbb{R}^2 \setminus \{B\}$. Pick an RBP trajectory ϕ starting at x and first reaching B at time τ, and let τ' be the supremum of those times $t \leq \tau$ for which $w(x) \leq t + w(\phi(t))$. Supposing $\tau' < \tau$, use the continuity of w to conclude that $w(x) \leq \tau' + w(\phi(\tau'))$. Note that $f\lceil(\mathbb{R}^2 \setminus \{|y| \leq 1/n\}) \times C$ is uniformly Lipshchitz for all n. Setting $\psi(\cdot) = \phi(\cdot + \tau')$ and $p = \phi(\tau')$ in the argument above and using the well-known local suboptimality principle for viscosity subsolutions for problems with uniformly Lipschitzian dynamics (e.g., Theorem III.2.32 of [1]), we get $w(\tilde{\psi}(1/n)) \leq \mu_2 - 1/n + w(\phi(\tau' + \mu_1))$ for large enough n, so $w(\phi(\tau')) \leq \mu_2 + w(\phi(\tau' + \mu_1))$. Hence, $w(x) \leq \tau' + \mu_1 + w(\phi(\tau' + \mu_1))$, which is a contradiction. Since $w(B) = 0$, it follows from infimizing that $w \leq T_B$ pointwise.

§4. References

[1] Bardi, M., and I. Capuzzo-Dolcetta, *Optimal Control and Viscosity Solutions of Hamilton-Jacobi-Bellman Equations*, Birkhäuser, Boston, 1997.

[2] Malisoff, M., "A remark on the Bellman equation for optimal control problems with exit times and noncoercing dynamics," to appear in *Proc. 38th IEEE CDC*.

[3] Sussmann, H.J., "Geometry and optimal control," in *Mathematical Control Theory*, J. Baillieul and J. C. Willems, Eds., Springer-Verlag, New York, 1998, pp. 140-198.

[4] Warga, J., *Optimal Control of Differential and Functional Equations*, Academic Press, New York. 1972.

Fault Protection in WDM Mesh Networks

Georgios Ellinas
Telcordia Technologies, 331 Newman Springs Rd., Red Bank, NJ.
{georgios@research.telcordia.com}

Abstract

This paper proposes a methodology for performing Automatic Protection Line Switching (APLS) to protect against a link failure in optical networks with arbitrary mesh topologies. In the scenario considered, the protection fibers' layout and the protection switches' set-up is implemented in non-real time, during the initialization of the network. When a fiber link fails, the connections that use that link are automatically restored and their signals are routed to their original destination using the protection fibers and protection switches.

1 Introduction

Optical communication systems are high-speed, high capacity systems that handle many types of user data services. These systems can transport vast amounts of information at any given time. Since fiber link failures are not uncommon for such systems, a fault recovery scheme that is fast (on the order of milliseconds) and reliable is needed. Using protection fibers and protection switches to perform Automatic Protection Line Switching will be the first line of defense against such failures [14].

APLS is commonly used in ring networks [6], to recover rapidly from link or switching node failures. In APLS systems, failures are circumvented by re-routing signals from"working channels" to "protection channels", using "protection switches" at the ends of each network link, which are activated immediately when a fault is detected. This work attempts to find sets of protection cycles, and to use these cycles to protect the traffic against link failures. This work is an extension of the ring cover approach presented in [7].

The networks considered here are composed of optical links and switches. Under normal operation, the network supports a number of active source-destination connections, whose paths are determined by the settings of the optical switches. A typical link consists of a pair of uni-directional working fibers accompanied by a pair of uni-directional protection fibers. Four 2×2 protection switches terminate the fibers. Under fault-free operation, these switches provide connections from the working fibers to the optical switches in the network nodes.

In the scenario proposed here, the interconnections among the protection switches before and after faults are "pre-computed" during the network set-up. In cases of link failure, connections using that link are automatically restored by re-routing them to their destinations using the protection fibers and protection switches to detour them around the fault.

In Section 2 we describe the general methodology for determining cycle covers in arbitrary mesh networks. Section 3 presents an alternative approach of obtaining protection cycles for (non-Eulerian) planar or non-planar networks and Section 4 explains how APLS protection works in the case of a fiber link failure. Conclusions follow in Section 5.

2 Cycle Covers in Arbitrary Graphs

Given an undirected graph G (with loops and multiple edges allowed), a *cycle cover* of G is defined as a set of (not necessarily distinct) cycles, $\mathcal{C} = (C_1, ..., C_m)$, $m \geq 1$, of G, such that each edge of G belongs to at least one cycle of $\mathcal{C}$. The *length* of a cycle is defined as the number of edges it contains. Given a graph G with cycle cover $\mathcal{C}$, the *cycle cover length* ($\mathcal{L}(\mathcal{C})$) is the sum of the lengths of all cycles in $\mathcal{C}$ [2]. In this method, additional protection fibers are required for the edges that are traced more than once in the cycle cover. Minimizing $\mathcal{L}(\mathcal{C})$ (i.e., minimizing the redundancy (in terms of protection fibers) in the network) is the primary focus of this approach.

The discussion in this section is limited to 2-edge connected graphs, since graphs with bridges have no cycle cover. The work distinguishes between Eulerian and non-Eulerian graphs. If the undirected graph G is Eulerian, its cycle decomposition S will provide a cycle cover with length $\mathcal{L}(\mathcal{C}) = |E(G)|$ (optimal solution). Algorithms exist that decide whether a graph is Eulerian and provide a cycle decomposition of an Eulerian graph. For the more general case of non-Eulerian graphs, two approaches are presented in this section in order to find a cycle cover : (1) a *planar subgraph* and (2) an *Eulerian transformation* approach.

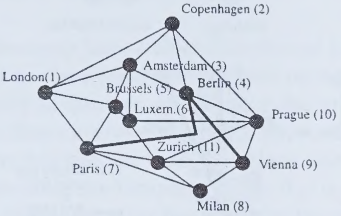
Planar Subgraph Approach:

This approach tries to obtain a cycle cover for an arbitrary graph G by first identifying a planar subgraph of a given arbitrary (non-planar) graph G, finding a cycle cover for the planar subgraph and then finding additional cycles for the "non-planar" edges. The problem of identifying a minimum number of "non-planar" edges (finding an optimal planar subgraph (a subgraph that results when a minimum number of edges is deleted) of a non-planar graph) is NP-complete [8].

A new, efficient (sub-optimal) graph-planarization algorithm was developed that attempts to find a number of non-planar edges which is very close to the minimum required for planarization. This algorithm is an extension of the Planarity Testing - Face Traversal algorithm presented in [5] and explained below. Even though this algorithm works for any arbitrary graph, it is particularly well suited for graphs that are *almost planar* (i.e., they have only a small number of non-planar edges).

This algorithm produces the plane embedding of the planar subgraph and a list of cycles corresponding to the faces of the plane embedding, as well as a number of non-planar edges. For the non-planar edges (single edges or walks), the algorithm obtains the shortest path between the two endpoints using Dijkstra's shortest path algorithm, and then constructs cycles using the shortest path and the non-planar edges. All cycles identified using this procedure will constitute a cycle cover.

This algorithm was tested for a number of complete graphs (where the minimum number of non-planar edges is known) and it was found to result in optimal planar subgraphs. The algorithm was also tested for a number of *almost planar* graphs (resembling telecommunications networks). In all cases, the number of non-planar edges obtained was extremely small. For example in Figure 1, which represents a *European* graph, 2 non-planar edges were found.



```
Number of faces found = 12

Face 1 : (2 1)(1 3)(3 2)
Face 2 : (2 3)(3 4)(4 2)
Face 3 : (2 4)(4 10)(10 2)
Face 4 : (1 7)(7 5)(5 1)
Face 5 : (1 5)(5 3)(3 1)
Face 6; : (3 5)(5 6)(6 3)
Face 7 : (5 7)(7 11)(11 6)(6 5)
Face 8 : (7 8)(8 11)(11 7)
Face 9 : (11 8)(8 9)(9 11)
Face 10 : (11 9)(9 10)(10 11)
Face 11 : (11 10)(10 6)(6 11)
Face 12 : (3 6)(6 10)(10 4)(4 3)

Non-planar edges : (4 9)(4 7)
(bi-directional)
```

Figure 1: Planar subgraph of a European network graph

Eulerian Transformation Approach:

This section presents a methodology for obtaining a cycle cover of non-Eulerian graph G by first transforming it to an Eulerian graph G_E. An optimal solution to this approach, based on a minimum weight perfect matching technique, and applied only to a class of graphs where their vertices can be covered by two edge-disjoint spanning trees was shown in [7]. This constraint limits considerably the application of the (Eulerian) optimized approach to a small number of graphs. A (sub-optimal) methodology that is applicable to any arbitrary graph is presented in this section.

A new algorithm was developed that transforms graph G to Eulerian graph G_E, performs a cycle decomposition of graph G_E and obtains a cycle cover for original graph G. Starting from a non-Eulerian graph G, all vertices with odd degrees are placed in set $V_{odd}(G)$. The edges that interconnect any two vertices v and u where v, $u \in V_{odd}(G)$ are placed in set $E_{odd}(G)$. Graph G is then reduced to graph $G_{odd}(V_{odd}, E_{odd})$. Each edge e in E_{odd}, joining two vertices of odd degree, is removed from G. If no edge exists between vertices of odd degree, an edge e is added between these two vertices (in G). Every time an edge e is added or removed, the two vertices at the endpoints of edge e have even degrees now and are not considered again. The procedure terminates when no more vertices of odd degree are left. The resulting graph, G_E, will always be Eulerian.

After a cycle decomposition S is performed for the Eulerian graph G_E, additional cycles are found for the edges previously added or removed. For each edge (i, j) that was previously removed, the shortest path from vertex i to vertex j with length > 1 (with all edge-weights set to 1) is found. Such a path always exists between any two vertices in G since it is always assumed to be 2-edge connected. The edges corresponding to the shortest paths are the ones that need to be added to the graph in order to create cycles for the edges previously removed. For each edge (i, j) previously added, an additional constraint is that the shortest path found cannot include any of the edges that are a part of the cycle which includes edge (i, j). This constraint is required to ensure that the cycle cover does not include cycles where an edge is traced twice. Such paths can always be found in a 2-edge connected graph where there are at least two edge disjoint paths between any two nodes v and u. Figure 2 shows an example of this algorithm.

The desired goal is to minimize the length of the cycle covers. However, nothing is known about the complexity of minimizing the length of a cycle cover, and it is conjectured that it is an NP-complete problem [11]. Itai in [11] showed that bridgeless, undirected graphs with n vertices and e edges can have *a* cycle cover with length at

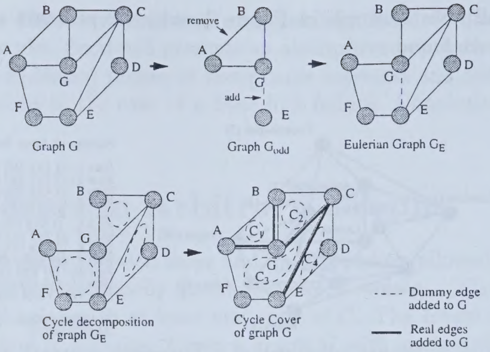


Figure 2: Transformation of graph G to eulerian graph G_E and cycle decomposition of graph G_E

most $\mathcal{L}(\mathcal{C}) \leq min\{3e-6, e+6n-7\}$. Bermond in [2], produced a tighter bound, $\mathcal{L}(\mathcal{C}) \leq e + min\{\frac{2}{3}e, \frac{7}{3}(n-1)\}$ ($\mathcal{L}(\mathcal{C})_{B_1}$), provided graph G has relatively few edges. For graphs that have two edge disjoint spanning trees, an even tighter bound exists, $\mathcal{L}(\mathcal{C}) = e + n - 1$ ($\mathcal{L}(\mathcal{C})_{B_2}$).

Table 1 compares the cycle cover lengths when the planar subgraph and Eulerian (optimal, sub-optimal) approaches are applied to the K_6, $\{5,5\}$-cage and European network graphs. In this table, $\mathcal{L}(\mathcal{C})_{PS}$, $\mathcal{L}(\mathcal{C})_{EO}$, $\mathcal{L}(\mathcal{C})_{ESO}$ denote the cycle cover lengths when the planar subgraph and Eulerian (optimal, sub-optimal) approaches are used respectively.

Graph Type	$2\mathcal{L}(\mathcal{C})_{PS}$	$2\mathcal{L}(\mathcal{C})_{EO}$	$2\mathcal{L}(\mathcal{C})_{ESO}$	$2\mathcal{L}(\mathcal{C})_{B_1}$	$2\mathcal{L}(\mathcal{C})_{B_2}$
K_6	32	36	42	50	40
$\{5,5\}$-cage	180	180	268	250	208
European	52	54	60	80	68

Table 1: Comparison of the cycle cover lengths for different graphs found using different cycle cover methods

The planar subgraph outperforms the rest of the proposed methods for these test cases. The optimal Eulerian approach also obtains small cycle cover lengths and it performs as well as the planar subgraph method for the $\{5,5\}$-cage graph. The sub-optimal Eulerian approach performs worse than both the planar subgraph and optimal Eulerian approaches but, contrary to the optimal Eulerian method it can be applied to any random graph. Values $\mathcal{L}(\mathcal{C})_{B_1}$ and $\mathcal{L}(\mathcal{C})_{B_2}$ are included in the table to demonstrate that some of the heuristics provide cycle covers with lengths that fall well below these bounds. In particular, the planar subgraph and Eulerian (optimal) approaches provide results that are better than these bounds for all test graphs.

3 General Methodology

Clearly, unless the graph representing the network is Eulerian, the number of protection fibers that are required using the approaches discussed in the previous section, will be more than the number of working fibers in the network. A new approach is discussed in this section that identifies cycle covers with minimal length.

The protection methodology proposed in this section, is based on connecting the protection fibers in a configuration that is a family of directed cycles, called *protection cycles*, which cover the network (digraph) in a particular manner.
The fundamental result is Proposition 1 :

Proposition 1: Recovery from a single link failure in any optical network with arbitrary mesh topology and bi-directional working and protection fiber links can be achieved using APLS, if a family of directed cycles using the protection fibers can be found so that : (i) all protection fibers are used *exactly once*, and (ii) in any directed cycle, a pair of protection fibers are not used in both directions unless they belong to a *bridge*.

Figure 3 (representing the complete graph K_5) is an example of a (non-planar) graph with four directed cycles covering the graph in the manner prescribed in Proposition 1. By appropriately interconnecting the protection switches in the corresponding network, these cycles would be implemented as protection cycles on the corresponding protection fibers.

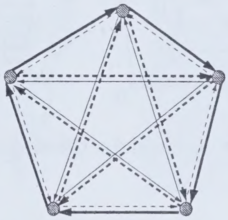


Figure 3: Directed cycles in the K_5 graph

3.1 Networks with Planar Topologies

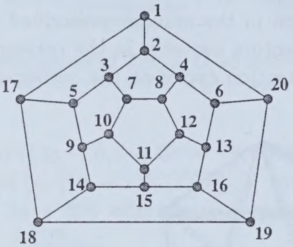
A graph is *planar*, if it can be embedded in the plane (or sphere). A graph G is said to be *embedded* in a surface S, when it is drawn on S so that no two edges intersect (have a common point other than a vertex) [1]. The resultant embedded graph is called a *plane* graph. The regions defined by the plane graph are called its *faces*, the unbounded region (on a plane) being called the *outer face*.

The protection cycles are then a set of facial cycles (f boundaries), with each one oriented properly. This result is embodied in Proposition 2 :

Proposition 2: Every planar graph G can be decomposed into a family of directed cycles where each edge is used exactly twice (once in each direction) and in each directed cycle that does not include a bridge an edge is used at most once. In each directed cycle including a bridge, the bridge is used twice (once in each direction) in the same directed cycle.

Thus, for planar graphs the required set of protection cycles can be obtained by embedding that graph on the plane, identifying the faces of the plane graph and traversing those faces in a certain direction. A new algorithm has been developed for this work that tests if a graph is planar and if it is, the algorithm embeds the planar graph in the plane and traces the faces in the appropriate directions to create a family of cycles with the characteristics defined in Proposition 1. The Planarity Testing - Face Traversal (PTFT) algorithm developed, is a variation of the path addition algorithm by Gibbons [9], which is itself a variation of the Demoucron, Malgrange and Pertuiset path addition algorithm [4]. The algorithm terminates if it determines that graph G is non-planar or if the number of faces f equals Euler's number (graph is planar). By adding one or more faces at a time, the graph is automatically embedded in the plane as well. All the edges are bidirectional and in the algorithm both directions of an edge are used in order to construct all faces. Thus, after the plane graph is constructed, the corresponding face tracing is also obtained.

The algorithm was tested for a variety of planar graphs and it was always able to obtain and traverse the corresponding face boundaries. Figure 4 shows an example of a planar graphs with the corresponding faces found using the algorithm.



```
Number of  faces = 12

Face[0] = [15 16][16 19][19 18][18 14][14 15]
Face[1] = [16 15][15 11][11 12][12 13][13 16]
Face[2] = [5 3][3 7][7 10][10 9][9 5]
Face[3] = [11 15][15 14][14  9][9 10][10 11]
Face[4] = [20 6][6 4][4 2][2 1][1 20]
Face[5] = [2 4][4 8][8 7][7 3][[3 2]
Face[6] = [5 9][9 14][14 18][18 17][17 5]
Face[7] = [5 17][17 1][1 2][2 3][3 5]
Face[8] = [19 16][16 13][13 6][6 20][20 19]
Face[9] = [12 11][11 10][10 7][7 8][8 12]
Face[10] = [18 19][19 20][20 1][1 17][17 18]
Face[11] = [4 6][6 13][13 12][12 8][8 4]
```

Figure 4: Face traversal for the dodecahedron graph

3.2 Networks with Non-Planar Topologies

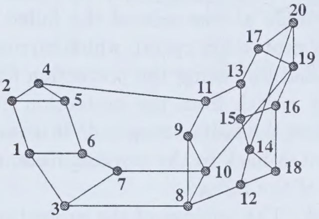
Extension of Proposition 2 to graphs with non-planar topologies requires certain results from graph theory dealing with "cycle double covers". A cycle double cover of a graph G is a cycle decomposition of G such that each edge appears in exactly two cycles. In the literature this problem is called the *cycle double cover problem* [12]. The set of protection cycles being sought comprise cycle double covers with some additional properties concerning their directions. A crucial conjecture concerning this problem is the following :

Cycle Double Cover Conjecture (CDCC) : Let G be a bridgeless finite graph. Then there exists a set of cycles in G such that each edge of G is in exactly two of the cycles (i.e., every bridgeless finite graph G has a cycle double cover).

A cycle double cover is said to be orientable when it is possible to choose a circular orientation for each cycle of the double cover in such a way that each edge is taken in opposite directions in the two incident cycles of the double cover [12]. An orientable cycle double cover is exactly our goal : a cycle decomposition such that each edge appears in exactly two cycles *and* each edge is used in opposite directions in the two cycles (i.e., the set of protection cycles is simply the family of cycles which are face boundaries in the strong orientable embedding that results in this orientable cycle double cover). Therefore, Proposition 2 applies to all graphs provided that an orientable cycle double cover exists for arbitrary graphs. Certain classical conjectures in graph theory suggest that this is the case [12]. For example, the face boundaries of the plane embedding of a planar graph (oriented properly) constitute an orientable cycle double cover (OCDC). Note that Figure 3 is also an example of an orientable cycle double cover of a non-planar graph.

While the cycle double cover conjecture has never been proven for arbitrary graphs, it was shown in [12] that a minimum counterexample[1] must be a *snark*. A snark is defined as a cyclically 4-edge-connected cubic graph of girth[2] at least five, which has chromatic index four. Furthermore, Celmins in [3] showed that the minimum counterexample to the CDCC must be a *strong snark* (a snark G such that for every edge e, $G * e$ (the unique cubic graph homeomorphic to $G - e$) is not edge colorable with 3 colors). Finally, it was further proven that a minimum counterexample to the cycle double cover conjecture has girth at least seven [10]. Thus, the minimum counterexample to the CDC conjecture has to be a strong snark of girth at least seven. But, no snark of girth at least seven is known to exist and it was conjectured in [13] that such snarks do not exist (conjectured that every snark has girth at most 6).

A heuristic was developed that finds the CDC of an arbitrary graph. This heuristic uses a *backtrack* approach to obtain the protection cycles of any graph. Figure 5 shows an example of this heuristic.



```
Number of Cycles = 10

Cycle[0] = [5 6][6 1][1 2][2 5]
Cycle[1] = [16 15][15 14][14 18][18 16]
Cycle[2] = [9 10][10 8][8 12][12 18][18 14][14 20]
[20 19][19 10][10 7][7 6][6 5][5 4][4 11][11 9]
Cycle[3] = [5 2][2 4][4 5]
Cycle[4] = [8 9][9 11][11 13][13 17][17 20][20 14]
[14 12][12 8]
Cycle[5] = [13 15][15 16][16 19][19 17][17 13]
Cycle[6] = [3 1][1 6][6 7][7 3]
Cycle[7] = [2 1][1 3][3 8][8 10][10 19][19 16][16 18]
[18 12][12 14][14 15][15 13][13 11][11 4][4 2]
Cycle[8] = [17 19][19 20][20 17]
Cycle[9] = [7 10][10 9][9 8][8 3][3 7]
```

Figure 5: Orientable cycle double cover of the Arpanet network

4 APLS Process Implementation

The previous section demonstrated how a family of protection cycles can be obtained, for networks with planar, and non-planar topologies. This section explains how these cycles can be used in order to protect any network with arbitrary mesh topology and bi-directional links against a single link failure.

4.1 APLS System Components

Each link consists of a pair of uni-directional working fibers (in opposite directions) and a pair of uni-directional protection fibers (in opposite directions). The protection fibers are the ones responsible for carrying the signals around a fault once a failure occurs.

The protection switches used to perform the APLS functions are 2×2 switches. These switches can be opto-mechanical switches or some other type (electro-optic, acousto-optic, thermo-optic, magneto-optic, etc.). Each bi-directional fiber link has four of these switches associated with it. During normal operation the switches are connected as follows : working port input (p_w^i) is connected to working port output (p_w^o) and protection port input p_p^i is connected to protection port output p_p^o. The ports of different protection switches are interconnected in a way that creates the family of protection cycles described in Section 3. These are calculated during the initialization of the network.

---

[1] If G is a minimum counterexample to the cycle double cover conjecture then G is a bridgeless graph with no double cover which has a minimum number of edges among graphs with these properties.

[2] Girth of a graph G is defined to be the order of the smallest cycle it contains.

4.2 APLS Process

The objective of the APLS process is to restore the connections passing through the failed fiber link once a failure occurs. That means routing the signals from the switching node at one side of the failed fiber link to the switching node at the other side of the failed fiber link using redundant protection fibers. Note that the APLS process will restore the connections that use the failed link in both directions.

As soon as the fiber link fails, the failure is detected in the switching nodes on both sides of the failed link. The protection switches then switch from working fiber to protection fiber and vice versa. The protection switches associated with the rest of the links in the same switching node remain at their default position. By switching the signals from the working fiber to the protection fiber the failed fiber link is automatically bypassed. When the signals come out of the switching node at one side of the failed link, they are switched onto a protection path (part of a protection cycle), which circumvents the failed link and reaches the other side of the failed link using the protection fibers. Once at the other side of the failed fiber link, they switch from the protection fiber to the working fiber, and follow the same path as before the failure occurred. It is important to emphasize that since the signals eventually switch back to the working fiber, the optical switches do not change their settings because of the failure.

Figure 6 shows a seven-node planar network. The settings of the protection switches (default settings) and a bi-directional connection from S to D are depicted. The protection ports at different protection switches are interconnected in such a way that the protection fibers associated with them form a family of directed cycles as defined previously in Section 3. Assuming a link failure occurs (Figure 7), the protection switches on both sides of the failed link switch from working to protection and vice versa. The path followed by the signals is shown by a solid dark line up to the failed link. The signals then switch to the protection fiber and follow the dotted dark line until they reach the other side of the failed link. There, they switch back to the working fiber and follow the same path as before the link failure.

5 Conclusions

This paper demonstrated a general methodology for performing APLS for link failures in optical networks with arbitrary mesh topologies and bi-directional links, where the number of protection fibers equals the number of working fibers in the network. It introduced the concept of *protection cycles* and developed algorithms to obtain these cycles in networks with planar and non-planar topologies. A complete solution was presented for planar and Eulerian networks. A key result was the association of the faces of the plane embeddings of planar graphs with the protection cycles and the development of a planarity testing, face traversal algorithm to obtain these cycles. The method was also extended to non-planar networks, and an orientable cycle double cover heuristic was developed that obtained the required family of protection cycles.

It also demonstrated the implementation of the APLS technique by showing that the protection switches located at the ends of the network links could be interconnected according to the identified protection cycles to ensure restoration from a link failure in a

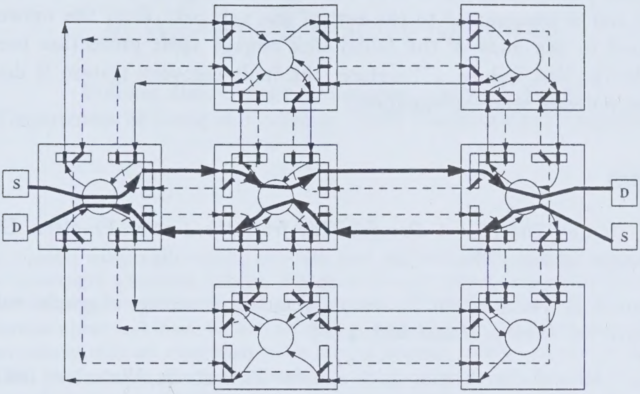


Figure 6: Seven-node planar network with default protection switch settings and a bi-directional connection

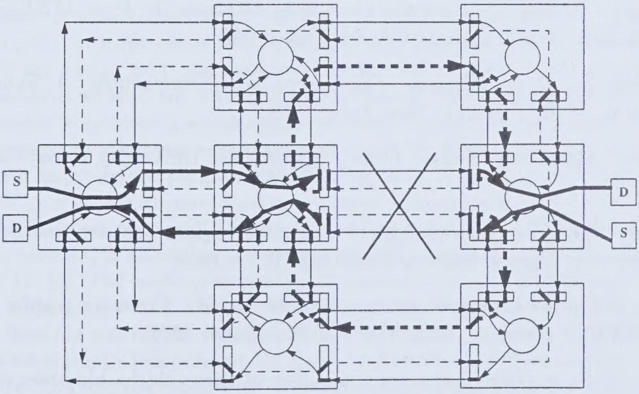


Figure 7: Restoration of a bi-directional connection in a seven-node planar network after a link failure

comprehensive, conflict-free system. This was achieved by re-routing signals from working to protection channels, using the protection switches, which were activated immediately after a fault was detected.

The restoration process is independent of the source-destination connections currently on the network and is transparent to the rest of the network. Only the network nodes that are attached to the ends of the failed link engage their protection mechanisms as a response to the link failure. Therefore, the fault recovery system is distributed, autonomous and network state independent.

References

[1] M. Aigner. *Graph Theory - A Development from the 4-color Problem.* BCS Associates, Moscow, Idaho, 1986.

[2] J. C. Bermond, B. Jackson, and F. Jaeger. Shortest coverings of graphs with cycles. *J. Combinatorial Theory*, 35:297–308, 1983.

[3] U. Celmins. *On conjectures relating to snarks.* PhD thesis, Waterloo, 1987.

[4] G. Demoucron, Y. Malgrange, and P. Peruiset. Graphes planaires. *Revue Francaise de Rech. Operat.*, 30:33–47, 1964.

[5] G. Ellinas, A. Hailemariam, and T. E. Stern. Planarity testing of a graph - Creation and traversal of faces. Technical Report CU/CTR/TR 474-97-07, Center for Telecommunications Research, Columbia University, New York, NY, 1997.

[6] A. F. Elrefaie. Multiwavelength survivable ring network architectures. In *Proc. IEEE Int'l Conf. Commun.*, pages 1245–1251, Geneva, Switzerland, May 1993.

[7] L. M. Gardner, M. Heydari, J. Shah, I. H. Sudborough, I. G. Tollis, and C. Xia. Techniques for finding ring covers in survivable networks. In *Proc. IEEE Globecom.*, pages 1862–1866, San Fransisco, CA, November 1994.

[8] M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide to NP-Completeness.* W.H. Freeman, New York, 1979.

[9] A. Gibbons. *Algorithmic Graph Theory.* Cambridge University Press, Cambridge, 1985.

[10] L. Goddyn. *Cycles in Graphs*, chapter : A girth requirement for the cycle double cover conjecture. Elsevier Science Publishers B.V., 1985.

[11] A. Itai, R. J. Lipton, C. Papadimitriou, and M. Rodeh. Covering graphs by simple circuits. *SIAM J. Comput.*, 10(4):746–750, November 1981.

[12] F. Jaeger. *Cycles in Graphs*, chapter : A survey of the cycle double cover conjecture. Elsevier Science Publishers B.V., 1985.

[13] F. Jaeger and T. Swart. *Combinatorics 79*, chapter : Conjecture 1. Elsevier Science Publishers B.V., 1980.

[14] T-H. Wu. *Fiber Network Service Survivability.* Artech House, 1992.

A Comparison of Allocation Policies in Wavelength Routing Networks [*]

Yuhong Zhu, George N. Rouskas, Harry G. Perros
Department of Computer Science, North Carolina State University

Abstract

We consider wavelength routing networks with and without wavelength converters, and several wavelength allocation policies. We show through numerical and simulation results that the blocking probabilities for the random wavelength allocation and the circuit-switched case provide upper and lower bounds on the blocking probabilities for two wavelength allocation policies that are most likely to be used in practice, namely, most-used and first-fit allocation. Furthermore, we demonstrate that using the most-used or first-fit policies has an effect on call blocking probabilities that is equivalent to employing converters at a number of nodes in a network with the random allocation policy. These results have been obtained for a wide range of loads for both single-path and general mesh topology networks. The main conclusion of our work is that the gains obtained by employing specialized and expensive hardware (namely, wavelength converters) can be realized cost-effectively by making more intelligent choices in software (namely, the wavelength allocation policy).

1 Introduction

Recent advances in wavelength division multiplexing (WDM) and optical switching make it possible to contemplate the deployment of wavelength routing networks that will provide backbone connectivity over wide-area distances and at very high data rates [7]. A wavelength routing network consists of wavelength routers and the fiber links that interconnect them. Wavelength routers are optical switches capable of routing a light signal at a given wavelength from any input port to any output port, making it possible to establish end-to-end lightpaths, i.e., direct optical connections without any intermediate electronics. The functionality of optical switches may be enhanced by employing wavelength converters, devices that are capable of shifting an incoming wavelength to a different outgoing wavelength [11]. Wavelength conversion is a desirable feature since it improves the performance of the network in terms of call blocking probability. However, this gain in performance must be weighted against the cost of wavelength converters.

The problem of computing call blocking probabilities under static (fixed or alternate) routing with random wavelength allocation and with or without wavelength converters has been studied in [1, 9, 2, 6, 12, 14]. The model presented in [1] is based on the assumption that wavelength use on each link is characterized by a fixed probability, independently of other wavelengths and links, and thus, it does not capture the dynamic nature of traffic. In [9] it was assumed that statistics of link loads are mutually independent, an approximation that is not accurate for sparse network topologies. In [2] a Markov chain with state-dependent arrival rates was developed to model call blocking in arbitrary mesh topologies and fixed routing; this technique was extended to alternate routing in [6]. While more accurate, this approach is computationally intensive and can only be applied to networks of small size in which paths have at most three links. A more tractable model was presented in [12] to compute recursively the blocking probabilities assuming that the load on link i of a path depends only on the load of link $i-1$. Finally, a study of call

---

[*] This work was supported by the NSF under grant ANI-9805016.

blocking under non-Poisson input traffic was presented in [14], under the assumption that link loads are statistically independent.

Other wavelength allocation schemes, as well as dynamic routing are harder to analyze. First-fit wavelength allocation was studied using simulation in [3, 9], and it was shown to perform better than random allocation, while an analytical overflow model for first-fit allocation was developed in [8]. A dynamic routing algorithm that selects the least loaded path-wavelength pair was also studied in [8], and in [10] an unconstrained dynamic routing scheme with a number of wavelength allocation policies was evaluated. Except in [12, 13], all other studies assume that either all or none of the wavelength routers have wavelength conversion capabilities. The work in [12] takes a probabilistic approach in modeling wavelength conversion by introducing the converter density, which represents the probability that a node is capable of conversion independently of other nodes in the network. Finally, in [13] a dynamic programming algorithm to determine the location of converters on a single path that minimizes average or maximum blocking probability was developed under the assumption of independent link loads.

Most of the approximate analytical techniques developed for computing blocking probabilities in wavelength routing networks [9, 2, 6, 14, 8, 10, 13] make the assumption that link blocking events are independent and amount to the well-known *link decomposition* approach [5]. Also, the development of some other techniques is based on the additional assumption that link loads are also independent. Link decomposition has been extensively used in conventional circuit switched networks where there is no requirement for the *same* wavelength to be used on successive links of the path taken by a call. The accuracy of these underlying approximations also depends on the traffic load, the network topology, and the routing and wavelength allocation schemes employed. While link decomposition techniques make it possible to study the qualitative behavior of wavelength routing networks, we believe that more accurate analytical tools are needed to efficiently evaluate the performance of these networks, as well as to tackle complex network design problems.

We have considered the problem of computing call blocking probabilities in mesh wavelength routing networks with fixed and alternate routing and random wavelength allocation in [16]. Unlike previous studies, we have developed an iterative *path decomposition* algorithm for analyzing arbitrary network topologies. Specifically, we analyze a given network by decomposing it into a number of single path sub-systems. These sub-systems are then analyzed in isolation using our algorithm for calculating the blocking probabilities in a single path in a wavelength routing network [15]. The individual solutions are appropriately combined to form a solution for the overall network. This process repeats until the blocking probabilities converge. Our approach accounts for the correlation of both link loads and link blocking events, giving accurate results for a wide range of loads and network topologies. Our algorithms can also compute call blocking probabilities in a mesh network where only a subset of arbitrarily selected nodes are capable of wavelength conversion.

In this paper, we study the blocking performance of several wavelength allocation policies for various network topologies and traffic patterns. We show that the most-used and first-fit policies have very similar call blocking probabilities for all calls in a network, regardless of the number of hops used by the calls. We also demonstrate that the random policy and the circuit-switched case (i.e., a system with a converter in each node), for which analytical solutions exist for networks of large size, provide lower and upper bounds on the call blocking probability under the first-fit and most-used policies. We also present results which indicate that the call blocking probabilities of the first fit and most-used policies is similar to that of the random policy when a number of converters is employed in the network.

In Section 2 we study a single path in a wavelength routing network, and in Section 3 we consider mesh network topologies. We conclude with a summary of our findings in Section 4.

2 A Single Path of a Network

We consider a single path of a wavelength routing network. A k-hop path consists of $k+1$ nodes labeled $0, 1, \cdots, k$, and hop $i, i = 1, \cdots, k$, represents the link between nodes $i-1$ and i. Each link in the path supports exactly W wavelengths, and each node is capable of transmitting and receiving on any of the W wavelengths. We assume that calls arrive as a Poisson process. Let $\lambda_{ij}, j \geq i$, denote the arrival rate of calls that use hops i through j of the path, i.e., calls that originate at node $i-1$ and terminate at node j. If the request can be satisfied, an optical circuit is established between the source and destination for the duration of the call. Call holding times are exponentially distributed with mean $1/\mu$. Also, let $\rho_{ij} = \lambda_{ij}/\mu$ denote the offered load of calls using hops i through j.

We define a "segment" of a k-hop path as a sub-path consisting of one or more consecutive links of the original path. We let n_{ij}, $j \geq i$, be a random variable representing the number of calls using hops i through j that are currently active. We also let $f_{ij}, j \geq i$, be a random variable representing the number of wavelengths that are free on all hops i through j. We shall see shortly that random variables n_{ij} and f_{ij} are part of the state description of the Markov process corresponding to the k-hop path.

Some of the nodes in the path can be equipped with a wavelength converter. These nodes can switch an incoming wavelength to an arbitrary outgoing wavelength. If no wavelength converters are employed in the path, a call can only be established if the *same* wavelength is free on all the links used by the call. This is known as the *wavelength continuity* requirement, and it increases the probability of blocking for calls using multiple hops. If a call cannot be established due to lack of available wavelengths, the call is blocked. On the other hand, if a call can be accommodated, it is assigned one of the wavelengths that are available on the links used by the call. If there are multiple wavelengths available, a wavelength allocation policy must be employed to select a wavelength for the call. Different selection policies lead to different call blocking probabilities. In this paper we investigate the following four wavelength allocation policies:

- *Random allocation:* a call is randomly assigned to one of the wavelengths that are available on all the links that will be used by the call.

- *Most-used allocation:* the wavelength that is already in use on the largest number of links in the path is assigned to the call; ties are broken arbitrarily. The objective of the policy is to keep more wavelengths available for calls traveling over long paths.

- *Least-used allocation:* the call is assigned to the wavelength that is currently used in the smallest number of links in the path, with ties broken arbitrarily. This policy results in wavelength fragmentation, leading to higher blocking probability for calls using long paths.

- *First-fit allocation:* the wavelengths on each link are given labels in a fixed order, and the call is assigned to the wavelength with the smallest label that is available on all the links it requires. The objective of this allocation scheme is to minimize wavelength fragmentation. As we shall show later, its performance is very close to that of the most-used policy, but it is easier to implement since there is no need to maintain information on the global use of wavelengths.

In a path with wavelength converters, the above allocation policies are used to assign a wavelength to the call within each segment of the path whose starting and ending nodes are equipped with converters. In addition to these wavelength allocation policies, we will also consider the following case:

- *Circuit-switched paths*: paths in which there are converters at all nodes. In circuit-switching, a call can be established as long as at least one wavelength (not necessarily the same one) is free on each of the links required by the call. Consequently, wavelength allocation is not an issue under circuit-switching.

In our study, we have used a number of different traffic load patterns to compare the four wavelength allocation policies against each other and against circuit-switching. These patterns are representative of the wide range of loading situations that one expects to encounter in practice, and can be found in [17]. To ensure that the results are comparable across the different patterns, the load values were chosen so that the total load is the same for all patterns.

2.1 Policy Comparison for a Single Path of a Network

We have shown in [15] that the evolution of a 2-hop path with random wavelength allocation can be characterized by the Markov process $(n_{11}, n_{12}, n_{22}, f_{12})$. The first three random variables in the state description provide the number of active calls between the three source-destination pairs in the path, and the last random variable gives the number of wavelengths that are free on both links of the path. The state transition diagram of this Markov process is shown in Figure 1 for $W = 2$ wavelengths, and it is straightforward to see that the process is not time-reversible [15]. By modifying a few of the transition rates of this process, we were able to derive a time-reversible Markov process with the same state space, which has a product-form solution. We have demonstrated in [15] that the blocking probabilities obtained through the product-form solution to the time-reversible Markov process are very close to the blocking probabilities obtained through the numerical solution to the original Markov process for a wide range of traffic loads.

Let us now consider the same 2-hop path with the most-used wavelength allocation policy. This policy can be modeled as a Markov process with the same state description as the random policy case, i.e., $(n_{11}, n_{12}, n_{22}, f_{12})$. The key difference is that, under the most-used policy, if $n_{11} > n_{22}$, then we know that there is at least one wavelength that is used on hop 1 but not used on hop 2. Thus, an incoming call that uses the second hop only will be assigned a wavelength that is already in use on the first hop, and will cause a transition to state $(n_{11}, n_{12}, n_{22} + 1, f_{12})$; similarly for $n_{22} > n_{11}$ and incoming calls using only the first hop. (Under the random wavelength allocation policy, the transition could be to either state $(n_{11}, n_{12}, n_{22} + 1, f_{12})$ or to state $(n_{11}, n_{12}, n_{22} + 1, f_{12} - 1)$ if the number of free wavelengths on both hops $f_{12} > 0$ and one of these wavelengths is assigned to the call.)

The state transition diagram of the Markov process for the most-used allocation policy is shown in Figure 2 for a 2-hop path with $W = 2$ wavelengths. Again, it is straightforward to verify that this Markov process is not time-reversible. Comparing to Figure 1, we note that despite having the same state space, the two processes differ in two ways. First, some of the transition rates are different; for instance the transition rate from state (0,0,1,1) to state (1,0,1,1) is equal to $\lambda_{11}/2$ for the random allocation, but λ_{11} for the most-used allocation. Second, some of the transitions are missing in the new Markov process. For example, there is a transition from state (0,0,1,1) to state (1,0,1,0) under random allocation in Figure 1, but there is no such transition in Figure 2. Furthermore, since there is a transition from state (1,0,1,0) to state (0,0,1,1) in Figure 2, but no transition in the reverse direction, it is not possible to obtain an approximate time-reversible process by simply modifying some of the transition rates, as we did for the random policy. Although we do not have an approximate product-form solution for the most-used allocation policy, the state space for a 2-hop path is small enough so that the solution to the Markov process can be obtained numerically.

Based on similar arguments, it can be determined that the least-used wavelength allocation policy can also be modeled by a Markov process with the state description $(n_{11}, n_{12}, n_{22}, f_{12})$. The state transition diagram for this process is shown in Figure 3, and it can be easily verified that the process is not time-reversible.

If a converter is placed at node 1 of a 2-hop path (the only interesting possibility in this case), the system becomes equivalent to a 2-hop circuit-switched path, and it can be described by

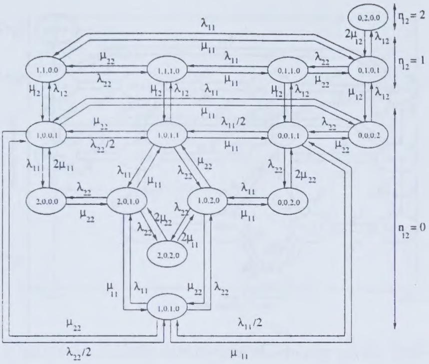


Figure 1: State space $(n_{11}, n_{12}, n_{22}, f_{12})$ of a 2-hop path with $W = 2$ wavelengths (random allocation)

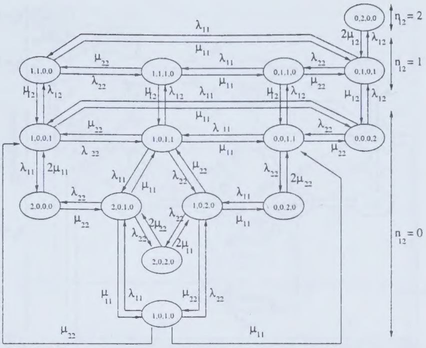


Figure 2: State space $(n_{11}, n_{12}, n_{22}, f_{12})$ of a 2-hop path with $W = 2$ wavelengths (most-used allocation)

the three-dimensional Markov process (n_{11}, n_{12}, n_{22}). Random variable f_{12} becomes redundant because calls using both hops can now use *any* of the $(W - n_{12} - n_{22})$ available wavelengths on the second hop. It is well-known that this process has a closed-form solution. In Figure 4 we show the state space of a 2-hop circuit switched path with two wavelengths. Although this path is described by the above 3-dimensional process, we include in the state description of Figure 4 the variable f_{12} to make it easier to compare to Figures 1–3. For instance, the fact that there are no transitions into state (1,0,1,0) in the figure can be explained by recalling that $f_{12} = 0$ (i.e., that no wavelength is free on both links of the path) implies that calls traversing both hops are blocked. However, since exactly one wavelength is free on each hop (even if it is not the same one), calls using both hops cannot be blocked in the circuit-switched path, and the system will never enter state (1,0,1,0), only state (1,0,1,1).

The first-fit wavelength allocation policy can also be modeled as a Markov process, but the size of its state space is in the order of W^5, too large to obtain a numerical solution even for relatively small values of W. In view of this, the blocking probabilities for this policy are obtained by simulation only.

2.1.1 Numerical Comparisons

Let us first consider the blocking probabilities of the random, most-used, least-used, and circuit-switched systems for calls traversing both links of the 2-hop path. In Figures 1 to 4, the blocking states for these calls are those with $f_{12} = 0$, i.e., those states in which neither of the two wavelengths is free on both links. We also observe that, except for state (1,0,1,0) at the bottom of each of the four figures, the transitions (and transition rates) in and out of all other blocking states are exactly the same for all four cases. Consequently, we expect that the difference in the blocking probability experienced by calls traversing both links of the path under the different policies will be mainly due to the steady-state probability of blocking state (1,0,1,0).

Referring to Figure 4, we note that the corresponding Markov process never enters state (1,0,1,0). Thus, we expect that calls traversing both hops will experience the least blocking probability in a circuit-switched path. In Figure 2 (most-used policy) we note that there are two transitions into state (1,0,1,0), and four transitions out of it. The blocking probability will be higher under this policy compared to the circuit-switched case. The Markov process in Figure 1 (random policy) has two additional transitions into state (1,0,1,0) from states (0,0,1,1) and (1,0,0,1) with rates $\lambda_{11}/2$ and $\lambda_{22}/2$, respectively. Therefore, the blocking probability of

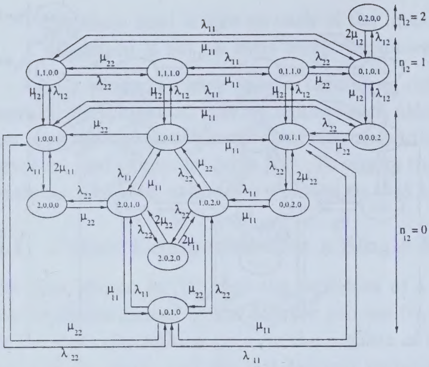


Figure 3: State space $(n_{11}, n_{12}, n_{22}, f_{12})$ of a 2-hop path with $W = 2$ wavelengths (least-used allocation)

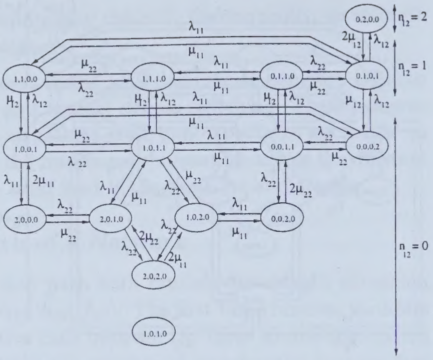


Figure 4: State space (n_{11}, n_{12}, n_{22}) of a 2-hop path with $W = 2$ wavelengths (circuit-switched)

these calls under the random policy will be higher than under the most-used policy. Finally, the Markov process in Figure 3 (least-used policy) has the same transitions as the one in Figure 1, but the transition rates into state (1,0,1,0) from states (0,0,1,1) and (1,0,0,1) are λ_{11} and λ_{22}, respectively. Therefore, we expect that these calls will experience the highest blocking probability under the least-used policy.

We now note that the lower the blocking probability for calls traversing both hops, the larger the number of such calls accepted, and the larger the number of wavelengths they occupy, thus leaving fewer wavelengths available for calls using a single link (either the first or the second) of the path. Hence, we expect that the behavior of the four policies in terms of the blocking probability of calls using a single link of the path will be exactly the opposite of what was discussed above. Specifically, we expect the least-used policy to provide the lowest blocking probability for these calls, followed by the random, the most-used, and the circuit-switched policies, in that order.

The above conclusions, derived by direct comparison of the states of the Markov processes, are in agreement with intuition. We have confirmed these conclusions by numerically comparing the blocking probabilities of the various policies for 128 different load values. Figures 5 and 6 show results for two cases corresponding to a uniform and descending load pattern, respectively, and for $W = 10$ wavelengths. More specifically, the arrival rates used to obtain the results in Figure 5 were $\lambda_{11} = 0.2, \lambda_{12} = 0.1, \lambda_{22} = 0.2$, while for the results in Figure 6 we used $\lambda_{11} = 3.0, \lambda_{12} = 2.0, \lambda_{22} = 2.0$. In both figures we plot the blocking probability for the three types of calls, namely, calls using the first hop only (label "hop 1" in the x-axis of the figures), calls using the second hop only (label "hop 2"), and calls using both hops (label "both hops"). We first note that the results are affected by the traffic pattern used. For instance, under uniform loading (Figure 5), calls using the first hop only experience the same blocking probability as hops using the second hop only, while in the descending pattern (Figure 6), due to the lower load offered to the second hop, the latter calls experience a much lower blocking probability for all four policies. More importantly, the relative values of the blocking probabilities for the four policies are also consistent with our discussion above. Very similar results have been obtained for all 128 different load values.

In Figure 7 we compare the most-used and first-fit policies for the same arrival rates as those used for Figure 6. We observe that the blocking probabilities of the first-fit policy are almost identical to those of the most-used policy for all three types of calls. This result can be explained by noting that both policies attempt to maximize the number of wavelengths that

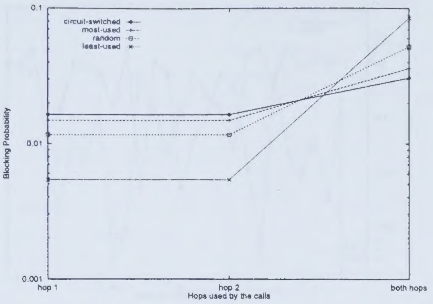


Figure 5: Policy comparison, 2-hop path, uniform traffic pattern

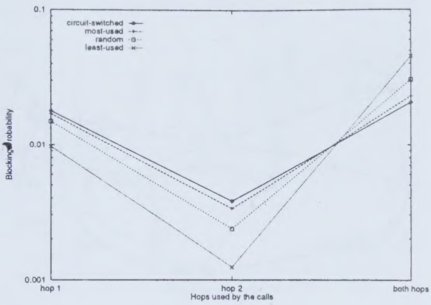


Figure 6: Policy comparison, 2-hop path, descending traffic pattern

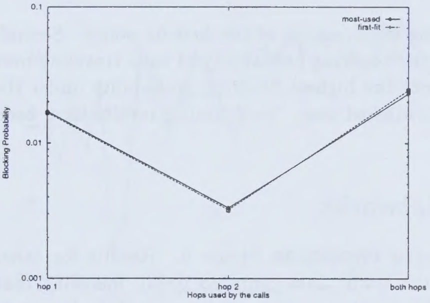


Figure 7: Most-used vs. first-fit allocation, 2-hop path, descending traffic pattern

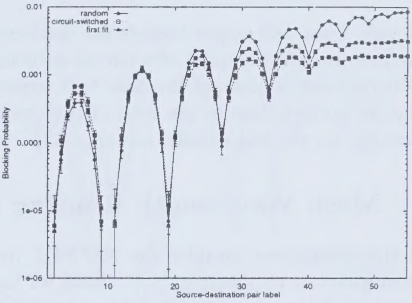


Figure 8: Policy comparison, 10-hop path, inverted bowl traffic pattern

are available for calls that use both hops of the 2-hop path by reducing the "fragmentation" of the set of wavelengths. The most-used policy assigns to an incoming call that requires a single hop of the path a wavelength that is already used on the other hop, if such a wavelength exists. On the other hand, the first-fit policy attempts to achieve the same goal by searching the set of wavelengths in a fixed order, thus increasing the chances that a wavelength used on a single hop will be assigned to an incoming call using the other hop. As can be seen from Figure 7, the most-used policy is slightly better, but overall the blocking probability values of the two policies are very close. Similar results have been obtained for all 128 traffic loads.

The relative behavior of the four policies for longer paths is very similar to shown in Figures 5 and 7 for a wide range of traffic patterns. Specifically, for calls using one or two hops of a path only, the least-used policy provides the lowest blocking probability, followed by the random policy, the most-used policy, and the circuit-switched case. However, for calls traversing three or more hops of the path, the situation is reversed. Due to space limitations, results for paths longer than two hops are omitted, but can be found in [17]. Since, under the least-used policy, the blocking probability of calls using multiple hops increases significantly, we will not consider the least-used policy any further.

Finally, in Figure 8 we compare the first-fit policy to the random (no converters) and circuit-switched cases for a 10-hop path (more results for 10-hop paths can be found in [17]). Two interesting observations can be made. First, the blocking probability values of the first-fit policy are always between the corresponding values of the random and circuit-switched cases. In other words, the blocking probability values under the random and circuit-switched cases

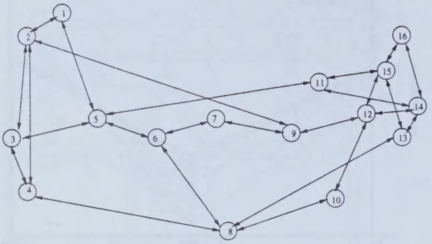


Figure 9: The NSFNET topology

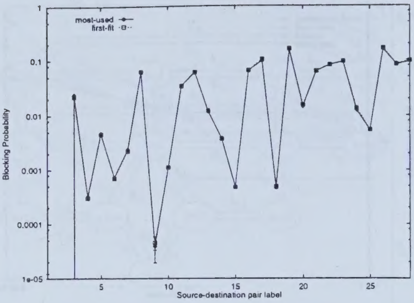


Figure 10: Most-used vs. first-fit allocation, NSFNET, pattern based on actual traffic

provide lower and upper bounds for the blocking performance of the first-fit policy. Second, the first-fit policy is quite effective in reducing the blocking probability of calls traveling over multiple hops (which are the ones that experience the highest blocking probability under the random policy) close to the level of the circuit-switched case. Very similar results have been obtained for the other traffic patterns [17].

3 Mesh Wavelength Routing Networks

In this section we consider the NSFNET irregular topology in Figure 9. Results for other topologies can be found in [17]. Since we use the traffic data reported in [4], following that study, we have augmented the 14-node NSFNET topology with nodes 1 and 16 in Figure 9. We present detailed results for the blocking probabilities of calls involving nodes along the path (3,5,6,7,9,12,15,16). There are 28 source-destination pairs in this path, and in Figures 10 to 14 they have been labeled so that numbers 1 to 7 refer to pairs with one-hop paths, numbers 8 to 15 correspond to pairs with two-hop paths, etc.

We have used two traffic patterns. For the first pattern, the call arrival rates are $\lambda_{sd} = 0.6 - l$, where l is the length of the shortest path from s to d. The second traffic pattern was designed to reflect actual traffic statistics collected on the NSFNET backbone network [4]. Clearly, this data, collected over a packet-switched network, cannot be directly applied to a circuit-switched wavelength routing network. However, our intention is simply to capture the relative traffic demands among the different source-destination pairs. To this end, we first divided the entries of the matrix in [4] by the link capacity to obtain the "offered load" ρ_{sd} per source-destination pair. Since the resulting values were too small, we multiplied them by a constant to obtain reasonable values for the offered load. Then, assuming that all calls have a mean holding time $1/\mu = 1$, the offered load values become the arrival rates λ_{sd} used in the experiments.

Figure 10 compares the first-fit to the most-used policies, and we again see that that the two policies result in almost identical blocking probability values for all calls. Figures 11 and 12 demonstrate that the random and circuit-switched cases provide upper and lower bounds on the performance of the first-fit policy, similar to the single-path cases studied above. Finally, in Figures 13 and 14 we compare the first-fit policy to the random policy with converters. The converters were placed in the network using the optimization techniques in [16]. As can be seen, using the first-fit policy is roughly equivalent to employing a significant number of converters in the network. The overall behavior of the graphs in these figures is very similar to the single path case as well as other topologies (see [17]), indicating that our observations and conclusions are valid for a wide range of topologies and traffic patterns.

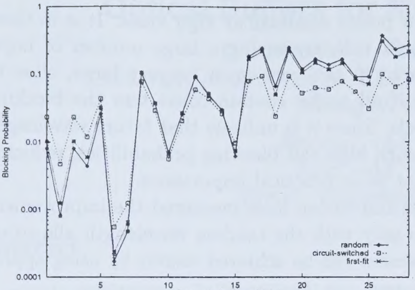


Figure 11: Policy comparison, NSFNET, traffic pattern based on locality

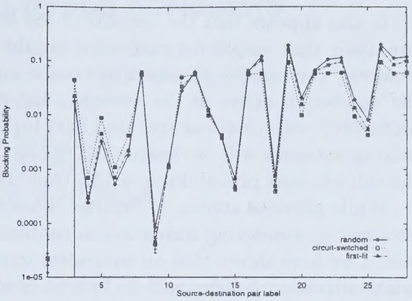


Figure 12: Policy comparison, NSFNET, pattern based on actual traffic

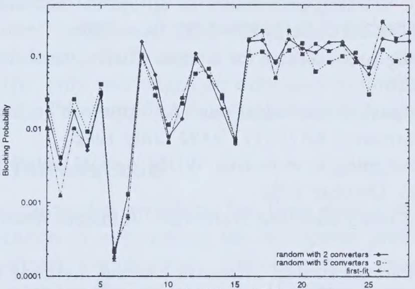


Figure 13: First-fit policy vs. random policy with converters, NSFNET, traffic pattern based on locality

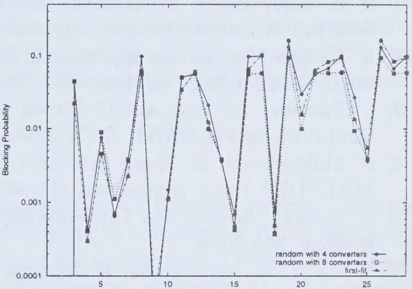


Figure 14: First-fit policy vs. random policy with converters, NSFNET, pattern based on actual traffic

4 Concluding Remarks

We have shown that the most-used and first-fit policies have very similar call blocking probabilities for all calls in a network, regardless of the number of hops used by the calls. The two policies tend to favor calls using multiple paths at the expense of calls using a single path. This is a desirable feature, since calls traversing multiple paths experience the highest blocking probability. However, the most-used policy requires that the network nodes exchange information about the network-wide usage of wavelengths, while the first-fit policy only relies on a fixed ordering of wavelengths, and is significantly easier to implement.

We have also shown that the random policy and the circuit-switched case provide bounds on the call blocking probability under the first-fit (or most-used) policy. Specifically, for calls using one or two hops, the random policy provides a lower bound and the circuit-switched case provides an upper bound, while for calls using longer paths the bounds are reversed.

We have presented results which indicate that the call blocking probabilities under the first-fit policy are similar to those under the random policy but employing a number of converters in the network. In most cases, introducing the first-fit policy results in a decrease in the blocking probability of calls traveling over multiple hops to a level very close to the blocking probability experienced under the circuit-switched case. Since, in terms of implementation, there is no significant difference between the first-fit and random policies, the gains obtained by employing expensive hardware can be realized by making more intelligent choices in software.

It also appears that the benefits of the first-fit policy diminish at high loads. It is in these situations that employing converters would benefit calls traversing a large number of hops. However, the number of converters to be employed in this case must be very large, close to the number of nodes in the network, and even if all nodes contain converters the blocking probability will remain at (reduced but) high levels. Since it is unlikely that future wavelength routing networks will be designed to operate at such high call blocking probabilities, reducing the call blocking probabilities in this case may not be of practical importance.

While previous studies of "sparse" wavelength conversion have measured the improvement obtained by employing converters in conjunction only with the random wavelength allocation policy, we have shown that an equivalent improvement can be achieved merely by using appropriate allocation policies such as first-fit or most-used.

References

[1] R. A. Barry and P. A. Humblet. Models of blocking probability in all-optical networks with and without wavelength changers. *IEEE J-SAC*, 14(5):858–867, June 1996.

[2] A. Birman. Computing approximate blocking probabilities for a class of all-optical networks. *IEEE J-SAC*, 14(5):852–857, June 1996.

[3] I. Chlamtac, A. Ganz, and G. Karmi. Lightpath communications: An approach to high bandwidth optical WANS. *IEEE Trans. Commun.* , 40(7):1171–1182, July 1992.

[4] B. Mukherjee *et al.* Some principles for designing a wide-area WDM optical network. *IEEE/ACM Trans. Networking*, 4(5):684–696, October 1996.

[5] A. Girard. *Routing and Dimensioning in Circuit-Switched Networks.* Addison Wesley, Reading, MA, 1990.

[6] H. Harai, M. Murata, and H. Miyahara. Performance of alternate routing methods in all-optical switching networks. *Proc. INFOCOM '97*, pp. 517–525, April 1997.

[7] A. Hill, A. Salek, and K. Sato (Eds.). Special issue on high-capacity optical transport networks. *IEEE J-SAC*, 16(7), Sep. 1998.

[8] E. Karasan and E. Ayanoglu. Effects of wavelength routing and selection algorithms on wavelength conversion gain in wdm optical networks. *IEEE/ACM Trans. Networking*, 6(2):186–196, April 1998.

[9] M. Kovacevic and A. Acampora. Benefits of wavelength translation in all-optical clear-channel networks. *IEEE J-SAC*, 14(5):868–880, June 1996.

[10] A. Mokhtar and M. Azizoglu. Adaptive wavelength routing in all-optical netowrks. *IEEE/ACM Trans. Networking*, 6(2):197–206, April 1998.

[11] B. Ramamurty and B. Mukherjee. Wavelength conversion in WDM networking. *IEEE J-SAC*, 16(7):1061–1073, Sep. 1998.

[12] S. Subramaniam, M. Azizoglu, and A. Somani. All-optical networks with sparse wavelength conversion. *IEEE/ACM Trans. Networking*, 4(4):544–557, Aug. 1996.

[13] S. Subramaniam, M. Azizoglu, and A. K. Somani. On the optimal placement of wavelength converters in wavelength-routed networks. *Proc. INFOCOM '98*, pp. 902–909, Apr. 1998.

[14] S. Subramanian, A. K. Somani, M. Azizoglu, and R. A. Barry. A performance model for wavelength conversion with non-poisson traffic. *Proc. INFOCOM '97*, pp. 500–507, Apr. 1997.

[15] Y. Zhu, G. N. Rouskas, and H. G. Perros. Blocking in wavelength routing networks, Part I: The single path case. *Proc. INFOCOM '99*, pp. 321–328, March 1999.

[16] Y. Zhu, G. N. Rouskas, and H. G. Perros. Blocking in wavelength routing networks, Part II: Mesh topologies. *Proc. ITC 16*, pp. 1321–1330, June 1999.

[17] Y. Zhu, G. N. Rouskas, and H. G. Perros. Bounds on the blocking performance of allocation policies in wavelength routing networks and a study of the effects of converters. Tech. Rep. TR-99-01, NCSU, Raleigh, NC, Jan. 1999.

Optical Buffers for Multi-terabit IP Routers

D. K. Hunter, I. Andonovic, M. C. Chia
University of Strathclyde, EEE Department, 204 George Street,
Glasgow G1 1XW, UK

Phone: +44 141 548 2527
Fax: +44 141 553 1955
Email: d.hunter@eee.strath.ac.uk

Abstract

A novel architecture is proposed for future multi-terabit IP (Internet Protocol) routers, employing multiple cascaded stages of optical switching and buffering. WDM is used within the node to facilitate its operation, to reduce buffering requirements, and also to transport signals between nodes. External synchronization is not required, and a void-filling algorithm is used to simplify hardware requirements. Packet priorities are not implemented in the current version of the switch, and the issue of header table lookup is not considered. Performance with respect to packet loss is studied by simulation; it is shown that for bursty traffic, only two stages are necessary to produce a packet loss of less than 10^{-6}, while for self-similar Pareto traffic, three stages are necessary.

1. Introduction

Optical packet switching has been extensively discussed in the literature recently, with the intention of overcoming the anticipated problems inherent in constructing future very large electronic switch cores. Although many architectures for optical packet switches have been proposed in the last few years [1], these have almost exclusively concentrated on fixed-length packets.

This paper proposes optical packet switched node architectures, which are suitable for use as IP (Internet Protocol) routers, since they switch and buffer variable-length packets optically. IP header address lookup will not be considered here, but although very high-speed IP address lookup has been demonstrated [2], it may be necessary to consider a simplified addressing scheme.

Each node may accept or transmit multiple packets simultaneously on each input or output fiber using WDM, enhancing its throughput, and also implying that the IP and WDM layers may be combined in future networks, simplifying network management and producing a highly flexible and functional packet switching layer.

Simulation is used to study the performance of the node under both bursty and self-similar traffic [3], the latter providing a meaningful comparison with real traffic.

Although switching elements of any size can in principle be constructed, those with 16 inputs and outputs are studied, which could be implemented optically and could form larger nodes with say 128 or 256 inputs and outputs, by means of a Clos architecture.

2. Principles of Node Design

High hardware complexity is a potential difficulty when implementing optical IP routers. Four measures are implemented to combat this:

- Asynchronous operation is permitted on router inputs, so it is not necessary to synchronize packets to timeslot or byte boundaries, prior to entering the switch. Synchronizers represent a considerable extra hardware burden and are expensive [4]. Here, it is assumed that packet lengths are multiples of one byte, while packet inter-arrival times are continuously distributed.
- Sharing of capacity among wavelengths is used to assist in contention resolution, reducing the buffering requirement [5,6,7]. If there is contention because two or more packets are directed to the same output simultaneously, initially an attempt is made to transmit each on a different wavelength. Buffers are still implemented in case the supply of wavelengths is exhausted. This greatly reduces the buffer requirement.
- If the lowest increment of delay permitted is less than the minimum packet size, packets may be directed in a FIFO manner to the appropriate outputs. However, the hardware complexity inherent in this approach may be avoided by using a technique, known as void filling, proposed by Tancevski [8]. Here, the delay-line increment may be much larger than the minimum packet length; as a result, unused gaps or "voids" are created between adjacent packets on any given output (Figure 1). Rather than scheduling a new packet under consideration for a particular output to be transmitted immediately after the last one, it may be scheduled to "fill" a void on the correct output in a non-FIFO manner. It is still possible to maintain FIFO operation for each input-output pair by stipulating that a packet cannot be scheduled for transmission before the previous packet between that particular input and output.
- By use of multiple buffer stages in cascade, the high splitting and combining losses, and the high hardware costs inherent in implementing only a single stage may be avoided. This approach has already been proposed for fixed-length optical packets in the Switch with Large Optical Buffers [9].

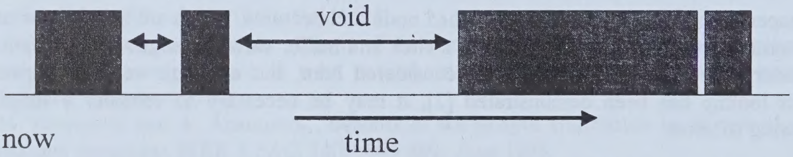


Figure 1: Formation of voids in the packet stream scheduled to pass from a given output. The void may be filled by scheduling another packet to be transmitted during it.

3. Node architecture

The architecture is composed of a series of S stages, numbered 0 to S-1 (Figure 2). Each stage has the following number of inputs and outputs:

- Stage 0: N inputs and D outputs.
- Stage S-1: D inputs and N outputs.
- All other stages: D inputs and D outputs.

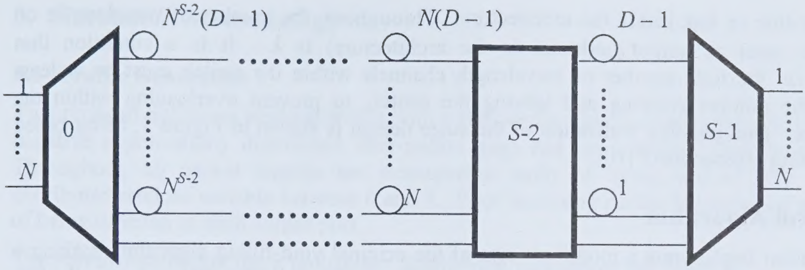


Figure 2: Outline of the switch architecture. It has *S* stages.

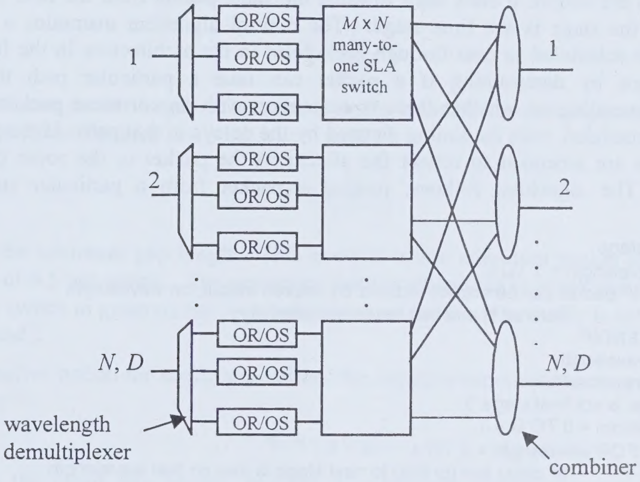


Figure 3: A "VWP" crossconnect proposed by NTT. In the packet switch proposed here, switching takes place at the packet rate. A packet entering on any input on any wavelength may be transmitted to any output on any wavelength. Wavelength converters may implement the "OR/OS" (optical receiver, optical sender) modules. There are *N* or *D* inputs and/or outputs, depending on the stage number

Each pair of adjacent stages are interconnected by delay-lines which can subject each packet to a delay in $\{0, \delta N^{S-1-i}, 2\delta N^{S-1-i}, ..., (D-1)\,\delta N^{S-1-i}\}$, with δ being the smallest unit of delay, known as the delay-line granularity. i denotes the stage before the delays, with $i = 0$ representing the leftmost stage, and N is the number of inputs and outputs. Each link or delay-line within the architecture can carry multiple packets at once, by means of wavelength division multiplexing.

Each packet entering a stage may leave on any other stage output, on any free wavelength. Hence each stage functions much like a crossconnect with wavelength conversion, only it operates at the packet rate. Λ is the maximum number of wavelengths per switching stage port, delay-line or link inside the architecture. Throughout, the number of wavelengths on each node input or output (external to the architecture) is λ. It is a condition that $D\Lambda \geq N\lambda$ i.e. the total number of wavelength channels within the switch must be at least equal to the number entering and leaving the switch, to prevent overloading within the architecture. One possible realization of the stage design is shown in Figure 3, being based on an existing crossconnect [10].

4. Control Algorithm

The algorithm implements a modified form of the original void-filling algorithm, making a sequential search of the available routes. Tancevski proposed void filling originally [8]. Packet priorities have not been examined, although this feature could be added.

The algorithms are recursive; each time the algorithm is called to route a packet from a certain stage to the output, it calls itself to route the same packet from the next stage to the output, unless the stage is the final stage. The control algorithm maintains a list of the packets that are scheduled to pass through each point in the architecture in the future. The algorithms work by determining if a packet can take a particular path through the architecture, depending on whether there is contention with one or more packets that have been already scheduled, with the timing dictated by the delays in that path. If the path is free, then these lists are amended to reflect the allocation the packet to the route through the architecture. The algorithm follows, routing a packet from a particular stage in the architecture:

```
IF stage is final stage
        FOR wavelength = 1 TO λ
                IF packet can be accommodated on desired output, on wavelength
                        reserve space and return successfully
                ENDIF
        NEXT wavelength
        return unsuccessfully
ELSE /* i.e. stage is not final stage */
        FOR linknum = 0 TO D – 1
                FOR wavelength = 0 TO Λ
                        IF delay line (or link) to next stage is free on that wavelength
                                /* recursive call to this function */
                                IF the packet can be routed from next stage
                                        reserve space and terminate successfully
                                ELSE
                                        GOTO X
                                ENDIF
                        ENDIF
                NEXT wavelength
X:      NEXT linknum
        return unsuccessfully
ENDIF
```

It is envisaged that the control unit will be implemented using VLSI technology. When the simulator itself calls the algorithm, routing takes place from the first stage of the architecture.

5. Simulation Methodology

5.1 Traffic Generation

Two types of traffic are studied: traffic with a self-similar Pareto distribution and traffic with negative exponentially distributed inter-packet gaps and geometrically distributed packets. Throughout, all packet lengths are measured in units of bytes, and U is a uniformly distributed random variable between 0 and 1. Each incoming packet has an equal probability of being directed to each output port.

The Pareto distribution has a probability distribution function or pdf given by:

$$f(t) = \alpha t_0^{\alpha} t^{-\alpha-1}$$

The inter-packet gaps are generated by the following formula, which can be derived from the pdf:

$$G = \frac{t_o}{U^{1/\alpha}}$$

t_0 is the minimum packet length. The same formula, truncated to an integer number of bytes, generates the packet lengths:

$$P = \left\lfloor \frac{t_o}{U^{1/\alpha}} \right\rfloor$$

Here, t_0 is the minimum gap length. This is equal to the minimum packet length, giving a traffic load of 0.5 per source. The necessary number of sources is employed before the first stage of the switch to generate the required volume of traffic. The traffic is self-similar for α between 1 and 2.

As an alternative option for traffic generation, the negative exponential inter-packet gaps are generated by:

$$G = -g \log U$$

where g is the mean gap length, and the geometrically distributed packet lengths are generated by:

$$P = \left\lfloor 1 + \frac{\log U}{\log(1 - 1/p)} \right\rfloor$$

where p is the mean packet length. A high mean packet length was chosen for the simulations, so that the distribution approximates to a truncated negative exponential, because $1 - 1/p \approx e^{-1/p}$ for large p. The packets have a minimum length of one byte.

5.2 Generation of Results

A simulator was written in C, and was exhaustively tested against existing results in the literature [6,7,8,11,12]. Simulations took place for 1.6×10^8 packets, to determine delay, and packet loss above 10^{-6}, with any smaller packet loss being considered acceptable. The loading level was 0.8 per wavelength throughout.

For negative exponential traffic, even with just two stages, 32 wavelengths throughout, 16 delay-lines per stage, and 16 inputs and outputs, the packet loss was always less than 10^{-6}.

Due to the deleterious effect of self-similar Pareto traffic, an appreciable packet loss was still obtained for two stages with such traffic. For Pareto traffic, the options selected were:

1. N (no. inputs/outputs) = 16
2. D (no. fiber delay lines per stage) = 16, 20, 24, … , or 64 (2 stages)
   $D = 16$ (3 stages)
3. S (no. stages) = 2 or 3
4. Λ (internal wavelength constraint) = 32, 48 or 64
5. λ (external number of wavelengths per port) = 32

Simulations were carried out using self-similar traffic [3], with $\alpha = 1.2$, mimicking real traffic [13], and corresponding to a Hurst parameter of 0.9. The minimum packet length was 400 bytes. The correct number of sources model the inputs, to provide the correct aggregate traffic level where each source provides a traffic level of 0.5. Each simulation was carried out with the basic delay-line unit δ equal to 400, 2000, or 5000 bytes.

6. Simulation under Self-Similar Traffic

6.1 Two-stage Architectures

Figures 4 and 5 show packet loss probability against number of delay-lines between stages, for delay-line granularities from 400 to 5000 bytes. For two stages, the number of internal wavelengths has little influence on performance, while the larger values of delay-line granularity produce superior performance. This is because the total amount of delay-line storage available is greater, and void-filling allows it to be utilized even although not all storage locations within it are immediately accessible.

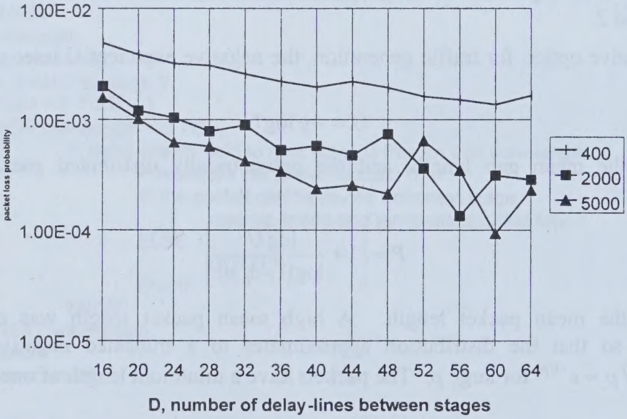


Figure 4: Packet loss probability under self-similar traffic for two stages with 32 internal wavelengths. The numbers in the box indicate the delay-line granularity.

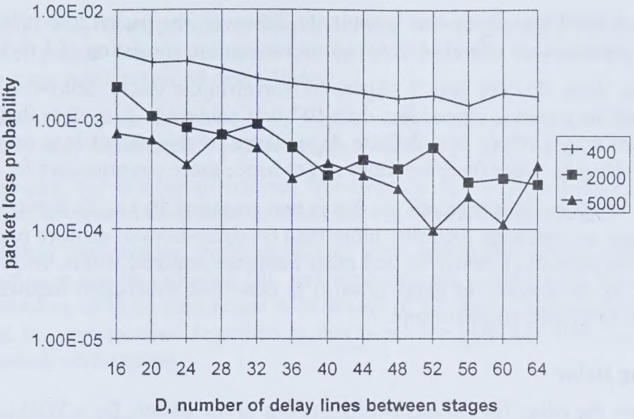


Figure 5: Packet loss probability under self-similar traffic for two stages with 64 internal wavelengths.

To reinforce the point that the number of internal wavelengths has little influence on performance, Figure 6 shows the packet loss probability against number of internal wavelengths, with $D = 16$ delay-lines between stages.

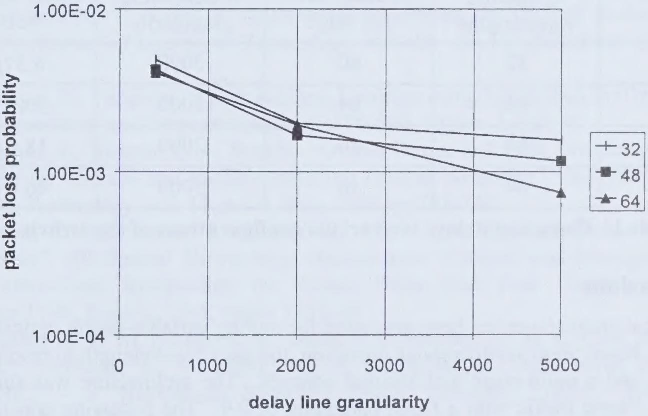


Figure 6: Packet loss under self-similar traffic for two stages with 16 delay-lines between stages.

6.2 Three-stage Architectures

Due to the simulation time involved, fewer simulations were carried out for three stages than for two. For 32 and 48 wavelengths, the packet loss is above 10^{-5} for all values of delay-line granularity. For 64 internal wavelengths, the situation is different. Here a granularity of 400 bytes yields a packet loss of 4.15×10^{-4}, this falling to 3.62×10^{-6} for a granularity of 2000 bytes. With a 5000 byte delay-line granularity, however, the packet loss is less than 10^{-6}, in fact, no lost packets were observed in the entire simulation, consisting of 1.6×10^{8} packets.

These results show that for just 3 stages, 64 wavelengths and a delay-line granularity of 5000, a satisfactory packet loss of less than 10^{-6} was achieved. It is clear that, unlike Figure 6 for just two stages, there is a definite dependence of the packet loss on the number of internal wavelengths, since there are more stages hence more opportunities for blocking.

The number of delay-lines required for this is two stages of 16 i.e. 32 in total. Contrast this with two-stage architectures requiring more than 64 delay-lines to achieve packet loss below 10^{-6}. Since the amount of switching and other hardware required within the router is roughly proportional to the number of delay-lines, it is clear that substantial hardware savings are possible with this multi-stage concept.

6.3 Queuing Delay

Table 1 shows the delay for various configurations of the switch, for a WDM channel bitrate of 40Gb/s. The results show that the price to be paid for a more economical architecture with more stages is increased delay. Also, the first two lines show that as the number of internal wavelengths is increased, the delay tends to decrease, since there is less contention within the switch. Also, predictably, increasing the size of the delay-line granularity in the last line increases the delay accordingly. Again, the penalty to be paid for superior packet loss performance appears to be increased delay.

$\|S\|$, stages	λ, internal wavelengths	D, delay-lines per stage	δ, delay-line granularity	mean queuing delay
2	32	60	5000	6.37μs
2	64	64	5000	3.43 μs
3	64	16	2000	18.9 μs
3	64	16	5000	46.4 μs

Table 1: Queueing delays for various configurations of the switch.

7. Conclusions

A new optical architecture has been presented for routing variable-length optical packets (e.g. IP packets), based upon asynchronous operation, the use of wavelength to resolve contention, void filling and a multi-stage architectural concept. The architecture was simulated under self-similar Pareto traffic with a Hurst parameter of 0.9. The following conclusions can be drawn from this work:

- With only 3 stages, when switching Pareto traffic, the architectures here can produce a useful packet loss of under 10^{-6}. This is because, by using multi-stage delay-lines, the switch is essentially emulating a large buffer, which becomes effective in combating the self-similar effects.
- Increasing the number of delay-lines between stages or increasing the number of stages improves the packet loss performance.
- Increasing the number of stages can reduce the overall number of delay lines and decrease the amount of hardware required to switch packets to the correct delay-lines.

For example, with three switching stages, having two stages of 16 delay-lines (32 in total), better packet loss performance can be achieved than is possible with a single delay-line stage having over 64 delay-lines.

- The performance of the architecture is improved by making the delay-line granularity much greater than the minimum packet length.
- For two stages, the number of internal wavelengths has little influence on the router performance, however it does have an effect for a greater number of stages.

Regarding opeitcal performance, previous studies have shown that several stages similar to those proposed here can be cascaded [9], and experimental studies have demonstrated the validity of cascading up to 40 such stages with the aid of all-optical regeneration [14]. Hence it is clear that the architectures described in this paper are practical with regard to optical noise and crosstalk performance.

8. Acknowledgments

The authors thank Ljubisa Tancevski of Alcatel Corporate Research Center, Richardson, for encouragement and useful discussions. David Hunter is funded by the EPSRC in the form of an Advanced Fellowship and Meow Chia acknowledges support from the Fujitsu Europe Telecom R&D Centre.

9. References

1. D. K. Hunter, M. C. Chia, I. Andonovic: "Buffering in Optical Packet Switches", *IEEE/OSA Journal of Lightwave Technology,* vol. 16, no. 12, December 1998, pp2081-2094
2. V. Srinivasan, G. Varghese: "Fast Address Lookups using Controlled Prefix Expansion", *ACM Transactions on Computer Systems*, 1999, vol. 17, no. 1, pp1-40
3. L. Tancevski, A. Bononi, L. A. Rusch: "Output Power and SNR Switching in Cascades of EDFAs for Circuit- and Packet-Switching Optical Networks", *IEEE/OSA Journal of Lightwave Technology*, vol. 17, no. 5, May 1999, pp733-742
4. A. Franzen, D. K. Hunter, I. Andonovic: "Low Loss Optical Packet Synchronizer Architecture", *All-Optical Networking: Architecture, Control and Management Issues.* SPIE International Symposium on Voice, Video and Data Communications, 1-5 November 1998, Boston, USA, paper 3531-40
5. D. K. Hunter, M. H. M. Nizam, K. M. Guild, J. D. Bainbridge, M. C. Chia, A. Tzanakaki, M. F. C. Stephens, R. V. Penty, M. J. O'Mahony, I. Andonovic, I. H. White: "WASPNET – a Wavelength Switched Packet Network", *IEEE Communications Magazine*, March 1999
6. S. L. Danielsen, B. Mikkelsen, C. Joergensen, T. Durhuus, K. E. Stubkjaer: "WDM Packet Switch Architectures and Analysis of the Influence of Tuneable Wavelength Converters on the Performance", *IEEE/OSA Journal of Lightwave Technology*, vol. 15, no. 2, February 1997, pp219-227
7. S. L. Danielsen, C. Joergensen, B. Mikkelsen, K. E. Stubkjaer: "Analysis of a WDM packet Switch with Improved Performance Under Bursty Traffic Conditions Due to Tuneable Wavelength Converters", *IEEE/OSA Journal of Lightwave Technology*, vol. 16, no. 5, May 1998, pp729-735
8. L. Tancevski, A. Ge, G. Castanon, L. Tamil: "A New Scheduling Algorithm for Asynchronous, Variable Length IP Traffic Incorporating Void Filling", *OFC '99*, San Diego, CA, February 1999

9. D. K. Hunter, W. D. Cornwell, T. H. Gilfedder, A. Franzen, I. Andonovic: "SLOB: a Switch with Large Optical Buffers for Packet Switching", *IEEE/OSA Journal of Lightwave Technology*, vol. 16, no. 10, October 1998, pp1725-1736
10. S. Okamoto, A. Watanabe, K. Sato: "A New Optical Path Crossconnect System Architecture Using Delivery and Coupling Matrix Switch", *Trans. IEICE Japan*, Oct. 1994, vol. E77-B, no. 10, pp1272-1274
11. M. G. Karol, M. J. Hluchyj, S. Morgan: "Input versus Output Queueing on a Space-Division Packet Switch", *IEEE Transactions on Communications*, vol. 35, December 1987, pp1347-1356
12. L. Tancevski, L. Tamil, F. Callegati: "Non-Degenerate Buffers: An Approach for Building Large Optical Memories", *IEEE Photonics Technology Letters*, vol. 11, no. 8, August 1999
13. W. Willinger, M. S. Taqqu, R. Sherman, D. V. Wilson: "Self-Similarity Through High-Variability: Statistical Analysis of Ethernet LAN Traffic at the Source Level", *IEEE/ACM Transactions on Networking*, vol. 5, pp71-86, February 1997
14. D. Chiaroni, B. Lavigne, L. Hamon, A. Jourdan, F. Dorgeuille, C. Janz, E. Grard, M. Renaud, R. Bauknecht, C. Graf, H. P. Schneibel, H. Melchior: "Experimental Validation of an All-Optical Network Based on 160Gbit/s Throughput Packet Switching Nodes", *24th European Conference on Optical Communications (ECOC '98)*, 20-24 September 1998, Madrid, Spain

Theλ-Scheduler: A Multiwavelength Scheduling Switch *

Jonathan P. Lang
Chromisys, Inc.,
421 Pine Ave.,
Goleta, CA 93117
jlang@chromisys.com

Emmanouel A. Varvarigos and Daniel J. Blumenthal
University of California,
Dept. of Electrical and Computer Engineering,
Santa Barbara, CA 93106
manos@ece.ucsb.edu
danb@ece.ucsb.edu

Abstract

We propose a new multiwavelength almost-all optical switch architecture called the λ-Scheduler that uses wavelength division multiplexing (WDM) internally to fold the switch architecture in both the space and time domains to reduce the hardware complexity and to improve the signal characteristics through the switch. The λ-Scheduler preserves the packet order for a given input-output pair, is consistent with virtual circuit switching, and when combined with appropriate connection and flow control protocols, provides lossless communication for *bursty* (or non-constant rate) traffic, provided the traffic satisfies certain smoothness properties. The λ-Scheduler uses novel scheduling and wavelength assignment algorithms, in conjunction with a series of feed-forward delay blocks, to avoid packet collisions within the switch or at the switch outputs. We present two implementations of the λ-Scheduler when the number of internal wavelengths k equal the number of inputs (and outputs) N to the switch. In the *Compressed* λ-Scheduler, the N internal wavelengths are used to fold the architecture in the time domain, which reduces the total number of delay blocks for the switch by $2N \log N$. In the *Collapsed* λ-Scheduler, the N internal wavelengths are used to fold the architecture in the space domain, which reduces the number of delay blocks and total fiber length used for delays by a factor of N.

1. Introduction

Fiber-optic communication links provide an extremely large (multi-THz) bandwidth potential with very low loss. Optical networks composed of *almost-all* optical switches, where the data packets remain in the optical domain and only the packet header or control information is processed electronically, can offer large bandwidth gains with extremely fast switching speeds while maintaining data transparency. This is because almost-all optical switches eliminate the need for optical to electronic to optical (O/E/O) conversion of the data, the so-called *electronic bottleneck*. Wavelength division multiplexing (WDM) can further exploit the large bandwidth potential of optical fiber by dividing the total fiber bandwidth into a number of narrower channels. As we will see, the flexibility provided by WDM can be used to fold switch architectures in the space and time domains, leading to a reduction in overall switch complexity.

Several researchers have proposed a number of optical switch architectures that were either single-wavelength systems (see, for example, [1], [2], [3], and [4]) or multi-wavelength (WDM) systems (see, for example, [5], [6], [7], and [8]). In the past, many researchers have argued that WDM-based systems were impractical for packet switching applications because wavelength tunable devices were too slow, or their filter characteristics were too broad, or they were simply too expensive. As the technology for wavelength tunable devices is improving, these arguments are losing strength, and researchers are beginning to reexamine WDM as an important facet of optical packet switching.

In this paper, we propose a new switch design, called the λ-Scheduler, which uses WDM internally to fold the switch architecture in both the space and time domains to reduce the number of elementary components and the total fiber length required to implement delays. The λ-Scheduler is based on the single-wavelength Scheduling Switch architecture of [4], and uses novel scheduling and wavelength

---

*Research supported by DARPA under the MOST project.

assignment algorithms, in conjunction with a series of feed-forward delay blocks, to avoid packet collisions within the switch or at the switch output, and to maintain the desirable properties of the original Scheduling Switch architecture. Two implementations of the λ-Scheduler are presented when the number of internal wavelengths k is equal to the number of inputs (and outputs) N to the switch. In the first implementation, called the *Compressed* λ-Scheduler, WDM is used to compress each branch of delay blocks and reduce the total number of delay blocks by $2N \log N$ compared to the Scheduling Switch. In the second implementation, called the *Collapsed* λ-Scheduler, WDM is used to collapse the N parallel branches of delay stages to a single branch, which reduces the number of delay blocks by a factor of N compared to the Scheduling Switch.

The organization of the remainder of the paper is as follows. In Section 2, we describe our objectives for the switch architecture and our assumptions on the traffic. In Section 3, we describe the single-wavelength Scheduling Switch architecture. In Section 4, we present the λ-Scheduler architecture and show how WDM can be used internally to fold the architecture in both the space and time domains. In Section 5, we describe a new component called the fast tunable grating reflector (FTGR) that provides the required flexibility to simultaneously switch multiple packets on different wavelengths independently. We present a cost analysis in Section 6 that compares two specific implementations of the λ-Scheduler, and we discuss the benefits of using WDM to fold the switch architecture.

2. Traffic Assumptions

The λ-Scheduler is designed to provide lossless communication for sessions that have certain smoothness properties, or sessions that can tolerate the delay induced when transforming them into smooth sessions. Following the discussion in [9], we view the time axis on a link as being divided into frames of length equal to T slots (see Fig. 1), where we assume that all packets have the same length and require one slot for transmission. A session is said to have the *(n, T)-smoothness property* at a node if at most n packets ($n \in \{1, 2, \dots, T\}$) of the session arrive at that node during a frame of size T. By using a leaky bucket scheme [10] to shape traffic at the source, and by ensuring frame integrity at the intermediate nodes, a session can be made to have the (n, T)-smoothness property throughout the network. This requires that the frame boundaries are maintained between the transmitting and receiving nodes, and can be done by setting a bit in the header to either 0 or 1, alternating values between frames.

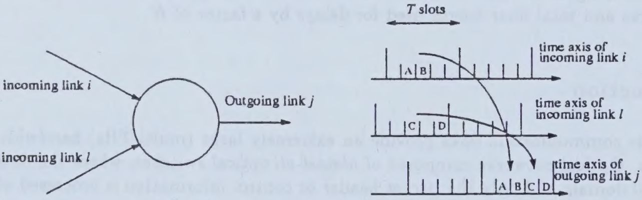

T slots
time axis of incoming link i
incoming link i
A
B
Outgoing link j
time axis of incoming link l
C
D
incoming link l
time axis of outgoing link j
A
B
C
D

Figure 1. We illustrate the incoming and outgoing frames at a node. Packets not intended for outgoing link j are not shown. Packets arriving in a particular frame of an incoming link that want to use the same outgoing link are sent over the same frame of the outgoing link.

The frame size T is an important parameter and can be viewed as a measure of the traffic burstiness we allow. The larger T is, the larger the flexibility in assigning rates to sessions. To see this, consider a network where each link has capacity C, and a session $\mathcal{S}$ has the (n, T)-smoothness property throughout the network. Session $\mathcal{S}$ will have an average rate of at most $R = nC/T$, and since capacity can only be allocated in discrete multiples of C/T, T determines the flexibility in assigning the rates.

We let N be the number of incoming (and outgoing) links of a node, and $n_{i,j}$, be the number of packets that arrive during a frame over incoming link i and have to be transmitted on the same outgoing frame of link j. Assuming unicast communication, we always have

$$\sum_{j=1}^{N} n_{i,j} \le T, \quad \text{for all } i \in \{1, 2, \dots, N\}. \tag{1}$$

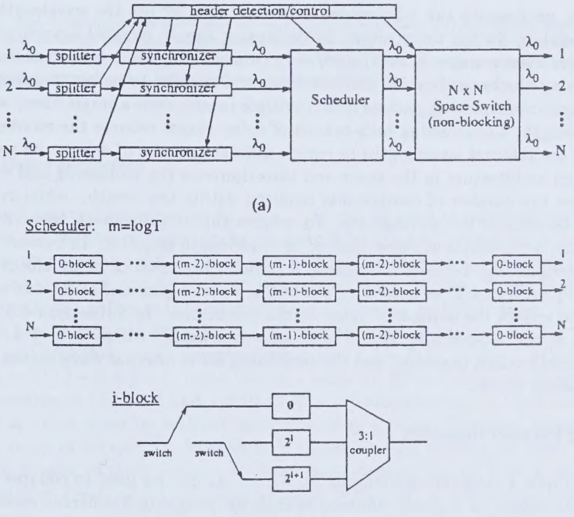
If the connection and flow control protocols used guarantee that the number of packets requiring the same outgoing link in a frame is less than or equal to the frame size T, i.e.

$$\sum_{i=1}^{N} n_{i,j} \leq T, \quad \text{for all } j \in \{1, 2, \ldots, N\}, \tag{2}$$

then all of the incoming packets can be assigned slots in the required outgoing links such that no packets will have to be dropped. Reservation based protocols, such as either *wait-for-reservation* (WR) protocols or *tell-and-go reservation* (TGR) protocols can be used to ensure lossless communication. (For a detailed discussion and comparison of WR and TGR protocols, see [11].) In practice, we expect the frame size T to be considerably larger than the number of inputs N. For example, if we have 16 $Gbps$ links and we want the basic rate of the sessions that are switched to be 16 $Mbps$, T will have to equal 1024.

3. The Scheduling Switch

In this section, we describe the Scheduling Switch architecture, first proposed in [4], which is used as the basic structure for the λ-Scheduler. The Scheduling Switch was designed to operate as a single-wavelength system, and is composed of a Scheduler followed by a $N \times N$ non-blocking Space Switch as shown in Fig. 2. At each input, a splitter is used for header detection, and packets are synchronized to a local clock so that the scheduling and switching operates synchronously. The purpose of the Scheduler is to rearrange the incoming packets so that packets appearing during the same slot at the outputs of the Scheduler require different outgoing links of the Space Switch. If this property is satisfied by the Scheduler, then the Space Switch will be able to route each packet to its desired outgoing link without any collisions or internal buffering.



(a)

(b)

Figure 2. (a) The Scheduling Switch architecture. (b) The Scheduler is implemented used N parallel branches, one for each input, where each branch consists of $2\log T - 1$ delay blocks.

An important data structure that is useful to describe the operation of the Scheduler is the *frame matrix*, defined as the $N \times N$ matrix $\mathcal{N} = \{n_{i,j}\}$, whose $(i,j)^{th}$ component is equal to the number $n_{i,j}, i = 1, 2, \ldots, N, j = 1, 2, \ldots, N$, of packets that arrive during a given frame $F(i)$ of incoming link i and require the same frame $F(j)$ of outgoing link j.

Definition 1: A *permutation matrix* $\mathcal{P}$ is any matrix with entries equal to "0" or "1" with the property that each line of the matrix has at most one nonzero entry.

The following lemma is found in [4].

Lemma 1: A frame matrix $\mathcal{N}$ can be written as the sum $\mathcal{N} = \sum_{s=1}^{T} \mathcal{P}_s$ of at most T permutation matrices, when condition (1) and (2) are satisfied.

Matrix $\mathcal{P}_s$ is used to determine the packet (if any) that will appear during slot s at each of the outputs of the Scheduler. In particular, if the $(i,j)^{th}$ entry of the matrix $\mathcal{P}_s$ is equal to one, then a packet arriving over link i and departing over link j is assigned to outgoing slot s of the Scheduler. Since $\mathcal{P}_s$ is a permutation matrix, this assignment guarantees that no packets arriving over the same incoming link or requesting the same outgoing link of the switch appear during the same outgoing slot s of the Scheduler. The first property ensures that there will be no collisions at the output of the Scheduler, while the second property ensures that there will be no collisions at the outputs of the space switch. Since the decomposition of the frame matrix into the T permutation matrices is not unique, we have the freedom to choose the order in which the packets are assigned to the outgoing slots of the Scheduler. In particular, this assignment can be chosen to preserve the order of packets arriving over the same input and requesting the same output.

The Scheduler is composed of N parallel branches, one for each input, where each branch delays the packets arriving on an incoming link until their assigned time slots are available on their desired outgoing links. This is equivalent to a time-slot interchanger and is implemented using $2\log T - 1$ three-state delay blocks, where we have assumed T is a power of 2. The i^{th} delay block can be implemented using two 2-state switches, one 3:1 coupler, and fiber delay lines corresponding to delays of 2^i and 2^{i+1} packet slots [see Fig. 2(b)]. The Scheduling Switch is a single-wavelength system that does not use WDM. In the next section, we present the λ-Scheduler, which is a scheduling architecture that uses WDM internally to reduce hardware complexity.

4. The λ-Scheduler

In this section, we describe the λ-Scheduler, and show how we use the wavelength flexibility of WDM to fold the Scheduling Switch architecture by collapsing the N parallel branches of delay blocks into fewer branches (or even a single branch) and/or by compressing each branch of delay blocks to reduce the number of delay blocks per branch. Collapsing several parallel branches to a single physical branch is done by wavelength multiplexing packets from multiple inputs onto a single fiber, where each input uses a different wavelength. Compressing each branch of delay stages reduces the number of delay blocks per branch by using the internal wavelengths to realize different groups of delays. As we will see, folding the Scheduling Switch architecture in the space and time domains (by collapsing and compressing the delay branches) reduces the number of components required within the switch, while preserving the original desirable characteristics of the architecture. To achieve this improvement, however, the delay blocks in each branch of the λ-Scheduler will now have to be wavelength selective. In Subsection 4.1, we show how k internal wavelengths can be used to collapse k parallel branches of delay blocks to a single common branch, and in Subsection 4.2, we show how k internal wavelengths can be used to compress each branch of delay blocks to reduce the number of delay blocks per branch. In Subsection 4.3, we present a general implementation of the λ-Scheduler with k internal wavelengths, where $w, w \leq k$, internal wavelengths are used to collapse parallel branches, and the remaining $k-w$ internal wavelengths are used to compress each branch of delay blocks.

4.1. Collapsing Parallel Branches

We first show how k internal wavelengths $\lambda_1, \lambda_2, \ldots, \lambda_k$ can be used to collapse a group of k parallel branches of delay blocks to a single common branch by assigning a different wavelength to each input (see also Fig. 3). Packets arriving on input $i, i = 1, 2, \ldots, k$, using wavelength λ_0 are first converted to wavelength λ_i using a fixed wavelength converter (WC_f). The packets are then multiplexed onto a single WDM fiber and transmitted along a single physical branch. In the original Scheduling Switch, the i-blocks in each of the k parallel branches are implemented using two simple 2-state switches that are independently set for each packet, followed by fiber delay lines with delays corresponding to 2^i and 2^{i+1} packet slots, and a 3:1 coupler. To obtain the same functionality using a single common branch and WDM, the simple 2-state switches of each delay block must now be implemented using multiwavelength selectable devices (e.g., FTGRs, to be discussed in Section 5) that can simultaneously switch packets on multiple wavelengths to different outputs, independently of each other. In other words, the i^{th} delay block of Fig. 3(b) will introduce a delay of $0, 2^i$, or 2^{i+1} packet slots depending on the wavelength on which each packet is carried and the desired delay for the packet. Once the packets have passed through all of the delay blocks on the single common fiber, they are demultiplexed back onto k parallel branches so that they can be switched in the Space Switch. Note, however, that if the demultiplexer is dynamic (in other words, if packets using different wavelengths can be arbitrarily routed to different outputs), then the dynamic demultiplexer can replace the Space Switch altogether.

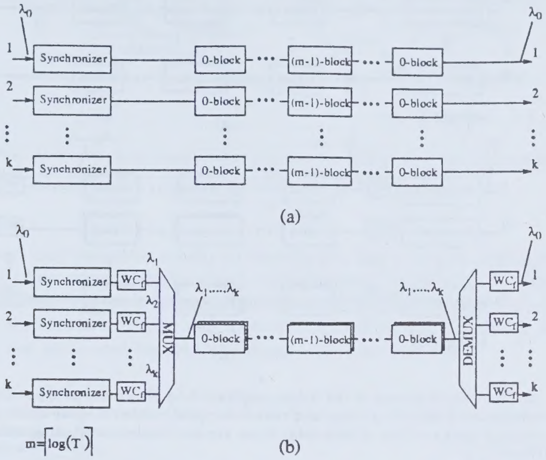

λ_0
λ_0
1
Synchronizer
0-block
(m-1)-block
0-block
1
2
Synchronizer
0-block
(m-1)-block
0-block
2
k
Synchronizer
0-block
(m-1)-block
0-block
k
(a)
λ_0
λ_0
λ_1
1
Synchronizer
WC$_f$
λ_2
$\lambda_1,\ldots,\lambda_k$
$\lambda_1,\ldots,\lambda_k$
WC$_f$
1
2
Synchronizer
WC$_f$
MUX
0-block
(m-1)-block
0-block
DEMUX
WC$_f$
2
λ_k
k
Synchronizer
WC$_f$
WC$_f$
k
$m=\lceil \log(T) \rceil$
(b)

Figure 3. (a) The delay branches implemented as k parallel branches. (b) WDM has been used to collapse the k parallel branches to single common branch using k wavelengths $\lambda_1, \lambda_2, \ldots, \lambda_k$. The number of delay blocks per branch remains the same as the single-wavelength system in (a), however, the simple 2-state switches of each delay block are now implemented using multiwavelength selectable devices.

4.2. Compressing Delay Branches

We now show how k internal wavelengths $\lambda_1, \lambda_2, \ldots, \lambda_k$ can be used to compress each branch of delay blocks, which reduces the number of delay blocks per branch by $2\lceil \log k \rceil$, where $\lceil x \rceil$ is the smallest integer greater than or equal to x (see also Fig. 4). The idea here is that each wavelength will be responsible for a smaller range of delays, which can be realized using fewer delay blocks. To see how this is done, we focus on an input link p and define the corresponding $T \times T$ *delay matrix* $\mathcal{D} = \{d_{i,j}\}$, where $d_{i,j}$ is the delay in packet slots that is required for a packet arriving on input link p in slot $i, i = 0, 1, \ldots, T-1$, of an incoming frame, and departing on an outgoing link in slot $j, j = 0, 1, \ldots, T-1$, of the outgoing frame, and is given by $d_{i,j} = 3T/2 - 2 + j - i$. Note that $d_{i,j}$ takes on $2T-1$ different integer values ranging from $\frac{T}{2} - 1$ (corresponding to $i = T-1$ and $j = 0$) to $\frac{5T}{2} - 3$ (corresponding to $i = 0$ and $j = T-1$), and it is these values of $d_{i,j}$ that must be realized using the delay blocks. Therefore, to reduce the number of delay blocks, the range of delays that have to be realized for each wavelength must be reduced.

To compress each branch of delay blocks, we map each element $d_{i,j}$ of $\mathcal{D}$ to a wavelength $\lambda_q, q = 1, 2, \ldots, k$, such that $q = 1 + (\lfloor \frac{i}{(T/k)} \rfloor - \lfloor \frac{j}{(T/k)} \rfloor) \bmod k$, where $\lfloor x \rfloor$ is the largest integer less than or equal to x, and we have assumed that T is a multiple of k (in practice, T and k will be powers of 2). This is equivalent to partitioning matrix $\mathcal{D}$ into k^2 smaller $\frac{T}{k} \times \frac{T}{k}$ submatrices $S_m, m = 1, 2, \ldots, k^2$, (indexed from left to right and top to bottom) such that the elements of each submatrix are mapped to a common wavelength, and the elements of adjacent submatrices are mapped to different wavelengths. Note that the values of $d_{i,j}$ in the submatrices with elements that are mapped to wavelength $\lambda_q, q = 1, 2, \ldots, k$, take integer values in the range

$$\frac{3T}{2} - \frac{qT}{k} - 1, \ldots, \frac{3T}{2} - \frac{qT}{k} - 3 + \frac{2T}{k}, \quad \text{for} \quad \left\lfloor \frac{i}{(T/k)} \right\rfloor \geq \left\lfloor \frac{j}{(T/k)} \right\rfloor \text{ and } q = 1, 2, \ldots, k,$$

and

$$\frac{5T}{2} - \frac{qT}{k} - 1, \ldots, \frac{5T}{2} - \frac{qT}{k} - 3 + \frac{2T}{k}, \quad \text{for} \quad \left\lfloor \frac{i}{(T/k)} \right\rfloor < \left\lfloor \frac{j}{(T/k)} \right\rfloor \text{ and } q = 2, 3, \ldots, k,$$

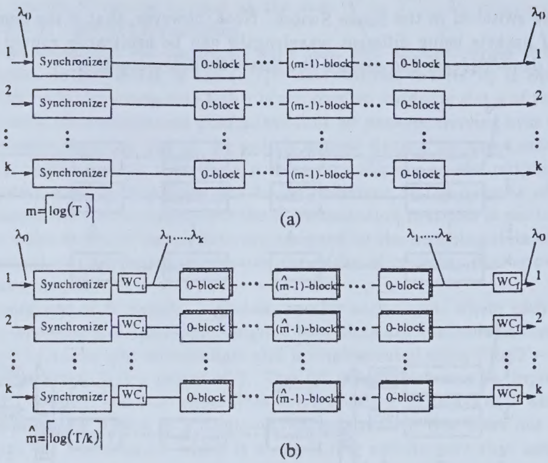

λ_0
λ_0
1
Synchronizer
0-block
(m-1)-block
0-block
1
2
Synchronizer
0-block
(m-1)-block
0-block
2
k
Synchronizer
0-block
(m-1)-block
0-block
k
$m=\lceil \log(T) \rceil$
(a)
λ_0
$\lambda_1,\ldots,\lambda_k$
$\lambda_1,\ldots,\lambda_k$
λ_0
1
Synchronizer
WC_t
0-block
$(\hat{m}-1)$-block
0-block
WC_f
1
2
Synchronizer
WC_t
0-block
$(\hat{m}-1)$-block
0-block
WC_f
2
k
Synchronizer
WC_t
0-block
$(\hat{m}-1)$-block
0-block
WC_f
k
$\hat{m}=\lceil \log(T/k) \rceil$
(b)

Figure 4. (a) The delay branches of the Scheduling Switch implemented as k parallel branches. (b) WDM has been used to compress each branch of delays and reduce the total number of delay blocks per branch by $2\lceil \log k \rceil$. As in Fig. 3, the simple 2-state switches of each delay block are now implemented using multiwavelength selectable devices (e.g., FTGRs).

and the number of different values that the $d_{i,j}$'s can take in each submatrix is $2(T/k) - 1$. Note also that the submatrices along the diagonals correspond to the same wavelength and have the same values for their elements, and two submatrices that correspond to the same wavelength but are not along the same diagonal have elements that differ by T.

Since the range of different values that the $d_{i,j}$'s can take in each delay submatrix is $2(T/k) - 1$, the delays of each submatrix can be implemented without collisions using a single branch consisting of $2\log(T/k) - 1$ delay blocks. (This can be seen by expanding the compressed branch of delay blocks using a space-time representation that realizes a Beneš structure, see [12].) The delays for submatrices corresponding to the same wavelength can be implemented using a common branch of delay blocks because of the non-blocking structure of each branch. The delays for submatrices corresponding to different wavelengths can also be implemented using a common branch of delay blocks, provided that multiwavelength selective devices are used for switching. For example, the fast tunable grating reflector (FTGR) that is discussed in Section 5 is a 2-state switch that can independently switch signals (packets) on multiple wavelengths to either output port. Therefore, the 2-state switches in the i-blocks of Fig. 2(b) can be replaced with FTGRs as shown in Fig. 5(a). Even though the number of values that the d_{ij}'s can take on in each submatrix is the same, an additional *offset* delay is required for each submatrix to account for the different values of the $d_{i,j}$'s. The offset delay can be separated into a fixed delay and a wavelength dependent delay [see Fig. 5(b)]. The fixed delay consists of a delay of $\frac{T}{2}(1 - \frac{1}{k})$ packet slots to account for the difference in minimum delays between a branch of $2\lceil \log T \rceil - 1$ delay blocks and a branch of $2\lceil \log(T/k) \rceil - 1$ delay blocks, and a delay of T packet slots to account for the difference in delays between submatrices corresponding to the same wavelength that are not along the same diagonal. The wavelength dependent delay of $(k - q)\frac{T}{k}$ packet slots accounts for the difference in delays between submatrices corresponding to different wavelengths $\lambda_q, q = 1, 2, \ldots, k$. The scheduling algorithm of Section 3 can be used in conjunction with the above wavelength assignment to switch packets to the outgoing slots without any packet collisions within the switch or at the switch outputs.

4.3. General Implementation of λ-Scheduler

The λ-Scheduler is a multiwavelength scheduling switch that is designed to rearrange the incoming packets so that packets appearing during the same slot at the outputs require different outgoing links. The key feature of the λ-Scheduler is that the scheduling and space switching are no longer restricted to a single wavelength; WDM is used instead, with a tunable wavelength converter (WC_t) at each switch

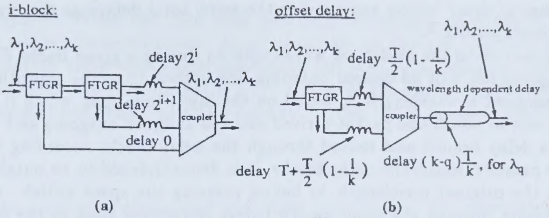

i-block:
offset delay:
$\lambda_1,\lambda_2,\ldots,\lambda_k$
FTGR
FTGR
delay 2^i
delay 2^{i+1}
delay 0
coupler
$\lambda_1,\lambda_2,\ldots,\lambda_k$
delay $\frac{T}{2}(1-\frac{1}{k})$
FTGR
wavelength dependent delay
coupler
delay $T+\frac{T}{2}(1-\frac{1}{k})$
delay $(k-q)\frac{T}{k}$, for λ_q
(a)
(b)

Figure 5. (a) The i-block, where the simple two-state switches of Fig. 2(b) have been replaced by multiwavelength switches or FTGRs (to be discussed in Section 5). (b) The offset delay block.

input, which can convert each packet to a different wavelength, and a fixed wavelength converter (WC_f) at each switch output, which converts packets back to the original wavelength. If wavelength conversion and fiber chromatic dispersion is used for synchronization ([13], [14]), we can combine the function of the synchronizer with that of the scheduler and space switch; folding the Scheduling Switch architecture will then require only one additional tunable wavelength converter per input.

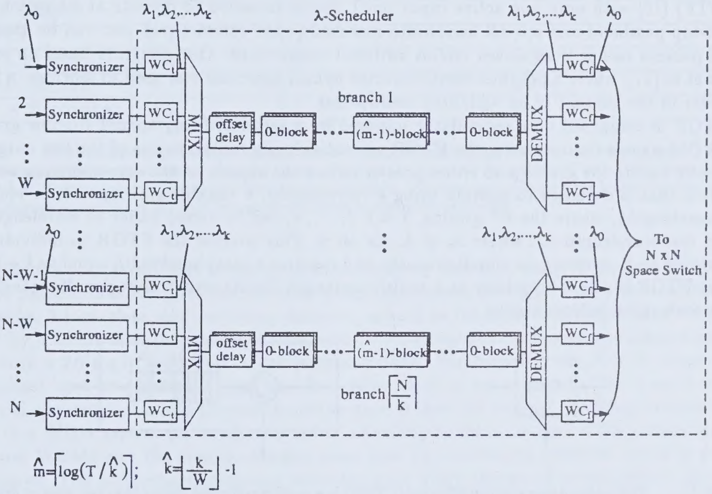

λ_0
$\lambda_1,\lambda_2,\ldots,\lambda_k$
λ-Scheduler
1
2
W
N-W-1
N-W
N
Synchronizer
WC_t
MUX
offset delay
0-block
$(\hat{m}-1)$-block
branch 1
branch $\lceil \frac{N}{k} \rceil$
DEMUX
WC_f
To N x N Space Switch
$\hat{m}=\lceil\log(T/\hat{k})\rceil$; $\hat{k}=\lfloor\frac{k}{W}\rfloor-1$

Figure 6. The λ-Scheduler uses WDM with a tunable wavelength converter (WC_t) at each switch input and a fixed wavelength converter (WC_f) at each switch output. We show an implementation with k internal wavelengths, where $w, w \leq k$, wavelengths are used to collapse parallel branches, and the remaining $k-w$ wavelengths are used to compress each branch of delay stages.

Let N be the number of inputs/outputs to the switch, k be the number of internal wavelengths (or channels) available, and T be the frame size in packet slots. In Fig. 6, we show a general implementation of the λ-Scheduler where $w, w \leq k$, internal wavelengths are used to collapse groups of w parallel delay-block branches into single branches, for a total of $\lceil \frac{N}{w} \rceil$ parallel branches. The remaining $k-w$ internal wavelengths are used to compress each branch of delay blocks, reducing the number of delay blocks per branch to $2\hat{m}-1$, where $\hat{m} = \lceil \log(T/\hat{k}) \rceil$ and $\hat{k} = \lfloor \frac{k}{w} \rfloor - 1$ is the number of wavelengths that are used to compress each channel of each branch of delay blocks.

The i^{th} delay block (or i-block) must now simultaneously switch packets using different wavelengths to one of three delay paths (corresponding to delays of $0, 2^i$, and 2^{i+1} packet slots), and is composed of two multiwavelength switches (FTGRs), fibers corresponding to delays of 2^i and 2^{i+1} packet slots, and one 3:1 coupler [see Fig. 5(a)]. An additional offset delay block [see Fig. 5(b)] is required so that the compressed branches of delay blocks can provide the same total delays as the original delay branches, as explained in Subsection 4.2.

A packet p that arrives to the switch on wavelength λ_0 during a given frame $F(i)$ of incoming link i is assigned an outgoing slot s of its desired outgoing link according to the algorithm given in Section 3. Packet p is then assigned a wavelength λ_q based on the input link upon which it arrived and the time delay between the slot in which the packet arrived and the assigned outgoing slot s. The packet is then multiplexed onto a delay branch and routed through the delay blocks according to the required delay. Once the packet p passes through the delay blocks, it is demultiplexed to an outgoing branch, and then converted back to the original wavelength λ_0 before entering the space switch. (It is also possible to wait until after passing through the space switch before converting back to the original wavelength so that WDM can also be used in the space switch implementation.)

5. FTGR

In this section, we describe a new multiwavelength component called the fast tunable grating reflector (FTGR), which is used to implement the i-block and offset delays for the λ-Scheduler [see Fig. 5(a) and 5(b), respectively]. Each WDM packet that arrives at the input of an i-block must be individually switched to one of three delay paths; similarly in the offset-delay block, each WDM packet must be individually switched to one of two delay fibers before passing through the wavelength dependent delay. The FTGR is functionally similar to the N-type building block of [15] [or to an acousto-optic tunable filter (AOTF) [16] with only one active input port], but is intended to operate at ns-switching speeds with a flat-top passband and a 3 dB bandwidth matched to the channel grid, and can be used to switch individual packets rather than slower circuit switched connections. Our design is based on recent work by Ozturk et al [17], where a polymer semiconductor hybrid approach was used to fabricate a high speed tunable filter in the context of an add/drop multiplexer.

The FTGR is composed of a recirculator followed by a series of finely spaced tunable gratings (see Fig. 7). WDM signals that arrive to the FTGR are individually routed to one of the two outgoing ports by electrically tuning the gratings to either pass or reflect the signals on the corresponding wavelengths. For a system that is designed to operate using k wavelengths, k tunable gratings will be required (one for each wavelength), where the i^{th} grating, $i = 1, 2, \ldots, k$, can be tuned either to wavelength λ_i or to a common *rest* wavelength λ_r, where $\lambda_r \neq \lambda_i$ for all i. This enables the FTGR to individually route signals arriving on k wavelengths simultaneously, and requires a total bandwidth equal to $k+1$ channels. We use the FTGR in the λ-Scheduler as a multiwavelength 2-state switch that simultaneously switches multiple wavelengths independently.

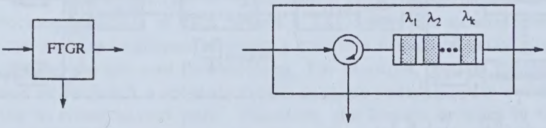


Figure 7. The fast tunable grating reflector (FTGR) is a multiwavelength device that can pass or drop signals on multiple wavelengths independently. The FTGR is composed of a recirculator followed by a a series of of tunable gratings that can be electronically tuned at ns packet rates to either pass or reflect the signals.

6. Cost Analysis

We now present a basic cost analysis, and compare the cost, in terms of the total number of elementary components, of the λ-Scheduler with that of the Scheduling Switch of [4]. Our purpose here is to study the benefits of using a multi-wavelength system instead of a single-wavelength system, as well as to study the tradeoffs between using WDM internally to collapse parallel branches or to compress each branch of delay stages.

Recall that N is the number of inputs/outputs to the switch and T is the frame size in packet slots, and we assume that $k = N$ is the number of internal wavelengths available. (In the following analysis, we will assume T is a multiple of N, and both T and N are powers of 2.) We will present a cost comparison between the Scheduling Switch and two implementations of the λ-Scheduler. In the *Compressed λ-Scheduler*, WDM has been used internally to compress each of the N branches of delay blocks, while in the *Collapsed λ-Scheduler*, WDM has been used internally to collapse all of the N parallel branches of the Scheduler to a single common branch. In all three switch designs, we have assumed that the synchronizer is implemented using wavelength conversion followed by a dispersive optical delay line as discussed in [13], the space switch (if required) is an $N \times N$ crossbar switch, and to simplify the figures, we have removed the splitter and header detection/control blocks, which are common to all three designs.

6.1. Component Cost

We first examine the number of elementary switching components, the number of wavelength converters, and the total fiber required to implement the delays (see also Table 1).

Components	Scheduler	Compressed λ-Scheduler	Collapsed λ-Scheduler
Delay Blocks	$2N \log T - N$	$2N \log(T/N) - N$	$2 \log T - 1$
Offset Blocks	—	N	—
Fiber Length (for delays)	$(9/2)NT - 6N$	$(13/2)T - 6N - T/N$	$(9/2)T - 6$
2-state switches	$4N \log T - 2N$	—	—
FTGR	—	$4N \log(T/N) - 2N + 1$	$4 \log T - 2$
3:1 Coupler	$2N \log T - N$	$2N \log(T/N) - N$	$2 \log T - 1$
2:1 Coupler	—	N	—
WC_t	N	2N	N
WC_f	N	N	2N
mux/demux	—	—	1
$N \times N$ crossbar	1	1	—

Table 1. Elementary component cost comparison between the Scheduling Switch, the Compressed λ-Scheduler, and the Collapsed λ-Scheduler, where N is the number of input/outputs to the switch, T is the frame size in packet slots, and $k = N$ is the number of internal wavelengths available. The cost measure for fiber length is given in terms of packet duration, and the other cost measures are given in component counts.

Comparing the component cost of the three structures, we find that using N internal wavelengths to collapse the parallel branches provides the best component count savings. This is because the number of delay blocks, 3:1 couplers, and switching elements, as well as the total fiber length used for delays are all reduced by a factor of N (compared to using the N wavelengths to compress the delay stages, which results in only a $2N \log N$ reduction in component count). Furthermore, the $N \times N$ crossbar switch can be replaced by a simple demux, and an offset delay block is not required. The main drawbacks of using internal wavelengths to fold the switch architecture is that the internal wavelengths cannot be used externally in a WDM system (ie, a separate set of wavelengths must be used if the incoming/outgoing links also use WDM) and the simple switches must now be wavelength selective which is potentially more expensive. The λ-Scheduler compares favorably with other almost-all optical switch architectures (see also Table 2) because it can provide lossless communication for bursty traffic with a hardware cost that grows logarithmically with the burst parameter T; it has a modular design; and WDM is used internally to share resources and reduce the number of elementary components.

References

[1] Z. Haas, "The 'Staggering Switch': An electronically controlled optical packet switch," *Journal of Lightwave Technology*, vol. 11, pp. 925–36, May/June 1993.

[2] M. Karol, "Shared-memory optical packet (ATM) switch," in *SPIE*, vol. 2024, pp. 212–22, July 1993.

[3] D. Hunter and D. Smith, "New architectures for optical TDM switching," *Journal of Lightwave Technology*, vol. 11, pp. 495–511, March 1993.

[4] E. A. Varvarigos, "The "Packing" and the "Scheduling" packet switch architectures for almost all-optical lossless networks," *Journal of Lightwave Technology*, vol. 16, pp. 1757–67, Oct. 1998.

Architecture	Lossless	Bursty	Modular	WDM	Elementary Components
Staggering [1]	No	No	No	No	$N \log b + b \log N$
SMOP [2]	No	No	No	No	$(N + b)^2$
BSS [6]	No	No	No	Yes	$bN + N^2$ $(b + 2N + 1)$ $(1 \times N)$-splitters $(N + 1)$ $(1 \times b)$-splitters
OASIS [5]	No	No	No	Yes	$N \log b$ b $(1 \times N)$-splitters
SLOB [8]	No	Yes	Yes	Yes	$kN \log b$ kb $(1 \times N)$-splitters
Packing [4]	Yes	Yes	Yes	No	$N^2 \log T + N \log N$
Scheduling [4]	Yes	Yes	Yes	No	$2N \log T + 2N \log N$
Compressed λ-Scheduler	Yes	Yes	Yes	Yes	$4N \log T$ FTGR
Collapsed λ-Scheduler	Yes	Yes	Yes	Yes	$N + 4 \log T$ FTGR

Table 2. We compare the λ-Scheduler with a number of other almost-all optical switch architectures when the number w of internal wavelengths is equal to the number N of input (and outputs) to the switch, T is the burst parameter, and b is the number of parallel delay branches in the single-stage switch architectures.

[5] J. Gabriagues and J. Jacob, "OASIS: A high-speed photonict ATM switch - results and perspectives," in *ISS'95*, pp. 457–61, April 1995.

[6] M. Renaud, F. Masetti, C. Guillemot, and B. Bostica, "Network and system concepts for optical packet switching," *IEEE Communications Magazine*, vol. 35, pp. 96–102, April 1997.

[7] S. L. Danielsen, B. Mikkelsen, C. Joergensen, T. Durhuus, and K. E. Stubkjaer, "WDM packet switch architectures and analysis of the influence of tunable wavelength converters on the performance," *Journal of Lightwave Technology*, vol. 15, pp. 219–27, February 1997.

[8] D. Hunter, W. Cornwell, T. Gilfedder, A. Franzen, *et al.*, "SLOB: a switch with large optical buffers for packet switching," *Jour. of Lightwave Technology*, vol. 16, pp. 1725–36, Oct. 1998.

[9] S. J. Golestani, "Congestion-free communication in high-speed packet networks," *IEEE Transactions on Communications*, vol. 39, pp. 1802–12, December 1991.

[10] A. Eckberg, D. Luan, and D. Lucantoni, "An approach to controlling congestion in atm networks," *Int. J. Digital and Analog Comm. Sys.*, vol. 3, pp. 199–209, April-June 1990.

[11] E. A. Varvarigos and J. P. Lang, "A virtual circuit deflection protocol," *IEEE/ACM Transactions on Networking*, vol. 7, pp. 335–49, June 1999.

[12] J. Lang, *Almost-all optical switching: Architectures, protocols, and analyses.* PhD thesis, UC Santa Barbara, 1999.

[13] D. J. Blumenthal, "An all-optical packet synchronizer with nanosecond reconfiguration rate and 5 nanosecond delay tuning range." preprint, 1998.

[14] L. Zucchelli, D. DiBella, G. Fornuto, *et al.*, "An experimental optical packet synchroniser with 100 ns range and 200 ps resolution," in *ECOC'98*, vol. 1.3, pp. 587–8, Sept. 1998.

[15] Y.-K. Chen and C.-C. Lee, "Fiber bragg grating-based large nonblocking multiwave cross-connects," *Journal of Lightwave Technology*, vol. 16, pp. 1746–56, October 1998.

[16] D. Smith, R. Chakravarthy, Z. Bao, *et al.*, "Evolution of the acousto-optic wavelength routing switch," *IEEE Journal of Lightwave Technology*, vol. 14, pp. 2044–51, Sept. 1996.

[17] C. Ozturk, Y. Byun, S. Sakamoto, and N. Dagli, "Integrated polymer/semiconductor grating reflector for WDM applications," in *LEOS'98*, vol. 2, pp. 5–6, Dec. 1998.

ON DIFFERENT ROUTING STRATEGIES IN TRANSPARENT ALL-OPTICAL NETWORKS

Ozan K. Tonguz

Telecommunications Research Group
Department of Electrical Engineering
State University of New York at Buffalo
Buffalo, NY 14260
USA

E-mail: tonguz@eng.buffalo.edu

Abstract— A good candidate for the backbone network of Next-Generation Internet (NGI) is transparent optical networks. In implementing such networks, one could pursue different routing strategies. This paper briefly reviews three such routing strategies and highlights their main capabilities and limitations from a performance standpoint.

1 Introduction

To accommodate continuously increasing demand for bandwidth and flexibility, transport networks are being enhanced by adding more fibers and switches, by increasing the bit rate per fiber (e.g., to 2.5 or 10 Gb/s SONET rates), and by upgrading the switches' size, throughput and functionality (e.g., from circuit switching to various forms of packet switching such as ATM). Recent advances and envisioned future capabilities of optical technology promise all-optical networks (AONs) capable of providing improved economy, flexibility and robustness, while still capable of making use of embedded fiber plant.

There are 3 general types of AONs, which, from a functionality point of view, can be classified as follows [1]: (1) Passive Optical Networks (PONs); (2) Transparent AONs; and (3)Ultra-High Speed AONs, each of which will now be briefly discussed.

PONs, traditionally, refer to all-optical networks using only passive components (e.g., fibers, directional couplers, star couplers, wavelength routers, multiplexers, and filters). The emphasis in such networks is on low cost and low (or no) maintenance, and the intended applications are fiber-in-the-loop and fiber-to-the-home. Thus economy and the "bury it and forget it" philosophy are prime objectives.

Transparent AONs resemble PONs in many ways, but, rather than economy, they stress flexibility, high-performance and local to global coverage. The main idea is to provide optical transparency enabling each of a large number of optical WDM channels to propagate from source to destination without optical-to-electronic data conversion within the network. The signaling format of each of the WDM channels can be virtually arbitrary, providing tremendous potential for flexibility and economic upgradability.

Ultra-high speed AONs take advantage of the tremendous speed of various optical phenomena to realize peak channel data rates that are much higher than can be generated, controlled or switched electronically. In such networks, communication is affected by on-off modulated, ultra high-rate (e.g., 100 Gb/s) time-division multiplexed (TDM) streams of short optical pulses or solitons. Virtually all network controls and signal switching functions are accomplished optically; i.e., these are truly all-optical networks. However, they are not optically transparent. It appears that such ultra-high speed TDM AONs may be limited to LANs and MANs.

This paper focuses on transparent AONs as the most promising class of AONs in providing flexible and high-performance broadband communications for Next-Generation Internet. To this end, the potential of different techniques such as deflection routing, hybrid store-and-forward, and wavelength conversion are briefly reviewed.

2 Performance of Multiwavelength Optical Cross-Connected Networks with Deflection Routing

Transparent cross-connected optical networks with deflection routing have recently become the focus of much research [2]-[9]. The main idea behind transparent networks is to modulate a lightwave carrier with data packets and let these optical packets travel from source to destination through a sequence of intermediate nodes without conversion to electronic form. Cross-connected topologies achieve higher throughput than linear topologies like buses and rings. In this paper two such topologies, Manhattan Street (MS) and Shufflenet (SN) topologies, will be considered.

2.1 Node Structure

The node is composed of a stack of submodules, one per wavelength. The wavelength from the input fibers are spatially demultiplexed and sent to the appropriate submodule for add/drop operations and switching. Packets from the submodules are finally multiplexed onto the output fibers. Fig. 1 shows the block diagram of the submodule described in [8].

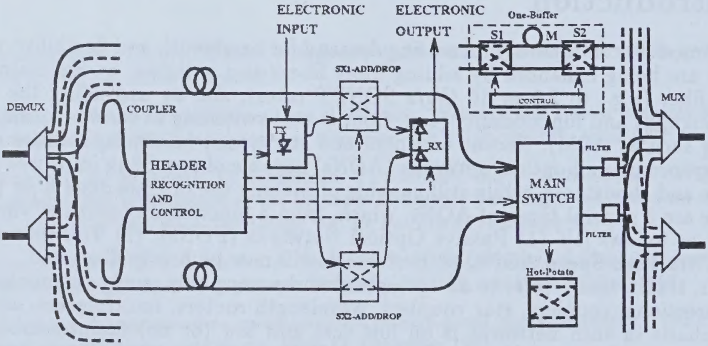


Figure 1: Node block diagram, Hot-potato switch, and One-buffer switch, M is the memory, S1 and S2 are exchange-bypass switches.

With hot potato routing the main switch is a simple crossbar switch. The scheme for a single-buffer memory is also shown in Fig. 1. A deflection occurs in the single-buffer scheme when three conflicting packets with the same output preference are present, two at the input links of the main switch and one in the buffer. The addition of more optical buffers reduces deflections but introduces more power losses and croasstalk. It will be shown that the presence of a single buffer can significantly degrade transmission performance.

2.2 Deflection Routing: Main Concept

Figure 2 shows a 16 node Manhattan Street Network with 2-connected nodes. Suppose nodes 9 and 6 both wish to communicate with node 1. Depending on the arrival times of the two packets from nodes 9 and 6, this scenario might lead to collision and, hence, blocking. To avoid blocking, one of the packets arriving at node 5 can be temporarily deflected to node 4. After going through node 5, the deflected packet will try again to reach node 1 by going through node 6 one more time. Hence, the main concept in deflection routing is *to use a fiber link as optical memory* in order to avoid conflict.

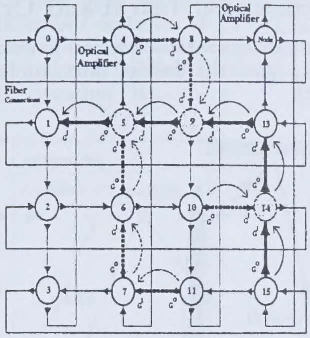


Figure 2: 16-nodes Manhattan street network.

2.3 Traffic Parameters

There are four parameters that determine the transmission performance [8]:

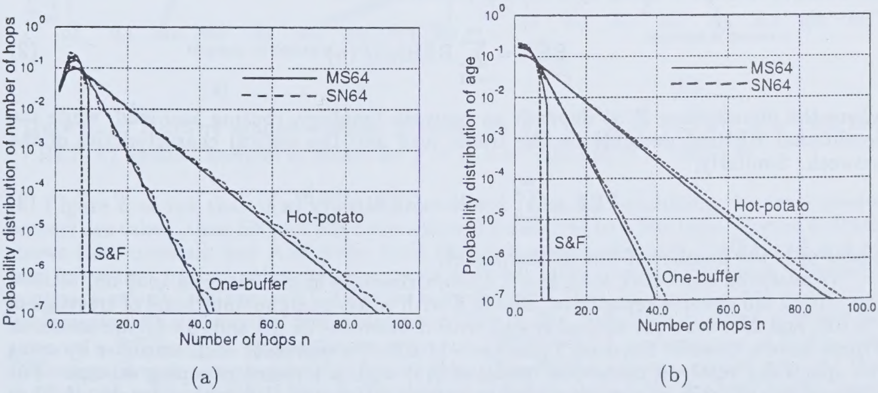
a)Average number of hops H that packets experience before absorption. The probability of packet absorption a is related to H as a=1/H.

b)The probability mass function (pmf) of the number of hops taken by a typical packet before absorption, shown in Figure 3(a).

c)The pmf of the age of packets, i.e. the number of hops experienced by a typical packet when it visits a generic node. This is depicted in Figure 3(b) and an analytic method to derive it is presented in [8]. Such curves are important when the amplifier gain is not equal to the interamplifier loss, since packets have power variations depending directly on their age. Curves for the average age A and average number of hops H versus the probability of packet generation g are also important [8].

d)The link utilization u is the probability of finding a packet at the input links of a node at each slot, when each submodule has two receivers and one transmitter, the link utilization is given as:

$$u = \frac{\sqrt{a^2 + g^2(1-a)^2} - a}{g(1-a)^2} \tag{1}$$



(a)

(b)

Figure 3: a) Probability distribution of number of hops versus n, b) Probability distribution of the age versus number of hops n.

2.4 Sources of Impairment: Intraband Crosstalk and ASE

The crosstalk generated in a 2X2 space switch is due to incomplete switching. A fraction 1-α of the signal power exits from the desired port, while a fraction α leaks from the undesired port (see Figure 4).

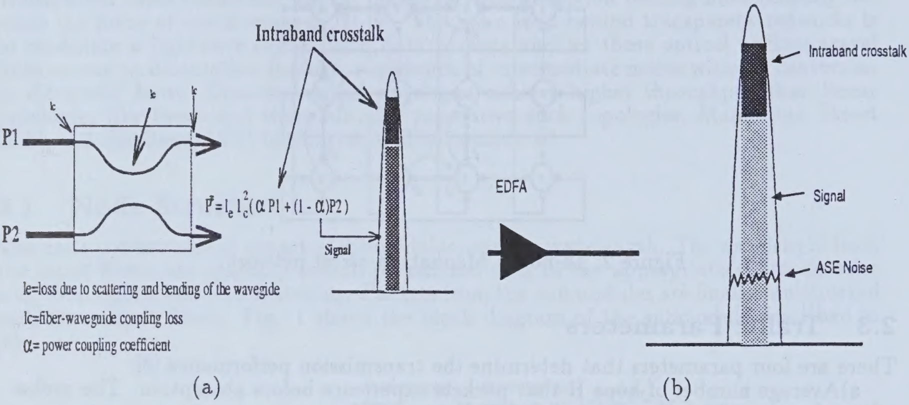

Intraband crosstalk
le
le
lc
P1
P2
$P=l_e l_c^2(\alpha P1+(1-\alpha)P2)$
Signal
le=loss due to scattering and bending of the wavegide
lc=fiber-waveguide coupling loss
α= power coupling coefficient
EDFA
Intraband crosstalk
Signal
ASE Noise
(a)
(b)

Figure 4: (a) Intraband crosstalk, (b) Amplifier spontaneous emmision (ASE) noise + intraband crosstalk.

If two signals at the same wavelength are present at the inputs, intraband crosstalk is generated.

The erbium-doped fiber amplifier (EDFA) used introduces amplifier spontaneous emission noise (ASE) in the band of interest which can not be eliminated by optical filtering. This is also shown in Figure 4. Thus, the two major sources of impairment that one has to deal with are ASE noise and intraband crosstalk.

2.5 Results and Discussion

A detailed analysis of Bit Error Rate (BER) and Packet Error rate (PER) are presented in [8], [9]. It is shown in [8] that the BER is obtained by conditioning the BER(n) on the number of hops n taken by a typical packet as

$$\mathrm{BER}=\sum_{n=1}^{\infty}\mathrm{BER}(n)P(n) \tag{2}$$

where the distribution $P(n)$ depends on network topology, routing and load, while the conditional BER(n) depends on the traffic load and the optical characteristics of the network. Similarly,

$$\mathrm{PER}=\sum_{n=1}^{\infty}1-[1-\mathrm{BER}(n)]^{m}P(n) \tag{3}$$

where m is the number of bits in the packet.

We analyzed a network with four 2.5 Gb/s channels in the range of 1550 nm to 1556 nm, with 2 nm channel separation, DEMUX with adjacent signal interband of crosstalk of -30 dB, and 2X2 crossbar optical switch with α between -25 dB and -35 dB are assumed. Filters have a transfer function $T(\Delta\lambda)=-17$ dB. We represent each amplifier by using the spectrally resolved numerical model of [10] with a forward pumping scheme. The absorption and gain parameters are the same as those of [10] with a length of 20 m and a pump power of 50 mW. Thus, it is assumed that the EDFAs are operating in the saturated regime. A bandwidth of 125 GHz is used to resolve the effect of ASE spectrum. The optical filter at the receiver has a 0.2 nm bandwidth and the electric filter has a 2.5

GHz bandwidth. We assumed a fiber with dispersion coefficient D_c=1 ps/km-nm, a loss coefficient of 0.2 dB/km, an internode distance of 15 km, a total node loss of -12.5 dB for hot-potato and -15.5 dB for single-buffer node architectures. The loss per single 2X2 switch is 3 dB. Two cases of the optical amplifier locations are analyzed: output amplifier location (A) and input amplifier location (B).

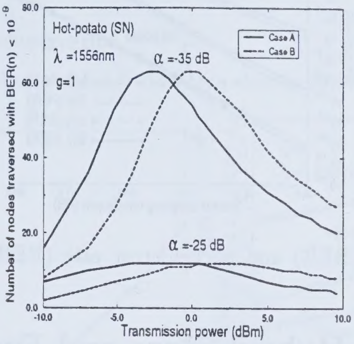
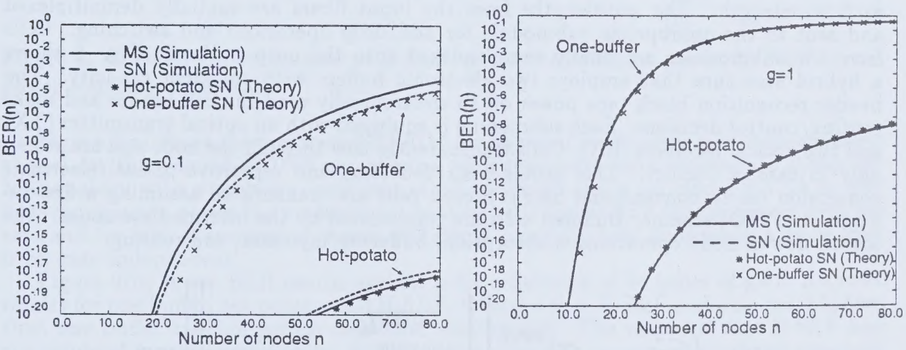


Figure 5: Maximum number of nodes traversed by a packet with a conditional BER(n) < 10^{-9} versus input power at g = 1, λ=1556 nm.



(a) (b)

Figure 6: a) BER(n) against number of nodes for $g = 0.1$, $\alpha = -35\ dB$, $\lambda = 1556\ nm$, b) BER(n) against number of nodes for $g = 1$, $\alpha = -35\ dB$, $\lambda = 1556\ nm$.

Figure 5 shows that if no special precautions (such as optimizing the transmission power) are taken, then SN and MS networks will be limited to a few hops. Figure 6 clearly shows that crosstalk and ASE noise limit the propagation distance, traffic load and bit rate in realistic optical networks. It appears that the most crucial component is the routing space switch (see Figure 7). Observe from Figure 7 that for practical power coupling coefficients (-25 dB) neither Hot-potato nor One-Buffer Deflection Routing schemes can support a BER=10^{-9}. While the transmission (BER/PER) performance of Hot-potato Deflection Routing is considerably better than One-Buffer scheme, results indicate that throughput performance of One-Buffer scheme is better (especially for high traffic loads) than that of Hot-potato [8], [9]. To resolve this dilema, in the next section Hybrid-Store and Forward (H-S&F) routing strategy will be investigated.

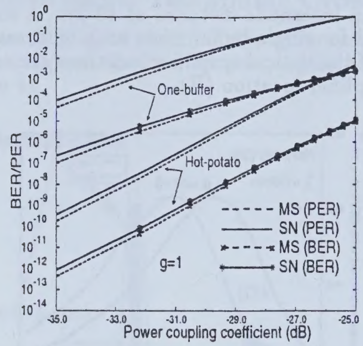


Figure 7: Bit error rate (BER) and packet error rate (PER) against power coupling coefficient α for $g = 1$.

3 Alternative: Hybrid-Store and Forward Topology

3.1 Node and Submodule Block Diagram

The node is composed of a stack of of independently-operated submodules, one per each wavelength. The wavelengths from the input fibers are spatially demultiplexed and sent to the appropriate submodule for add/drop operations and switching. Cells from the submodules are finally remultiplexed onto the output fibers. Fig. 8 shows a hybrid structure that employs two electronic buffers with sufficient capacity. The header recognition block taps power off to electronically read the cell header and make routing/control decisions. Each submodule is equipped with an optical transmitter (TX) and two optical receivers (RX). Cells transparently flow through the node and are stored only in case of conflict. This avoids both deflection and repetitive optical/electronic conversion (as in conventional S&F). Stored cells are transmitted assuming a first-in-first-out (FIFO) scheme. Buffered cells are regenerated by the intermediate nodes. The logical flow of node operations is absorption, buffering/injection, and routing.

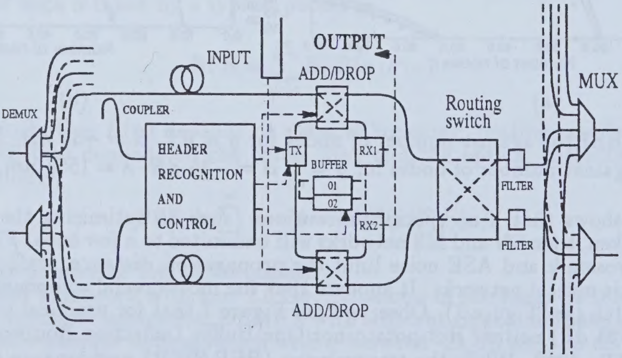
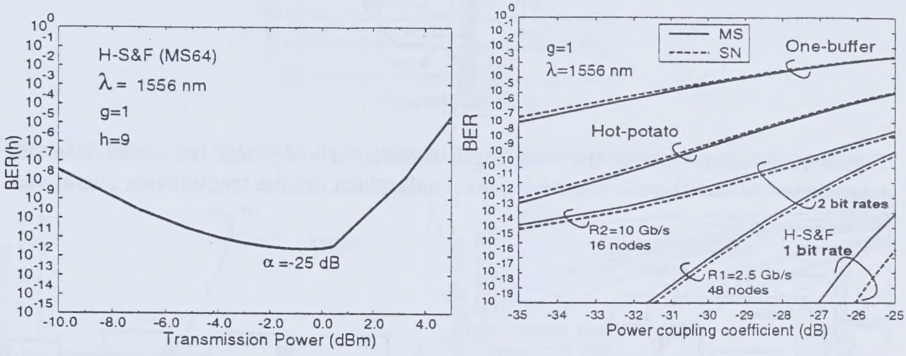


Figure 8: Node and submodule block diagram, submodule stores packets just in case of conflict.

When a cell is routed through a node, one of the two outputs is chosen according to a shortest path algorithm. Based on the position of the intermediate node and the cell's destination node, one or both outputs may be suitable for minimizing the number of hops a cell has to traverse for reaching destination. A cell that can take both outputs is called a "don't care cell", while a cell that has only one preferred output is called a "care cell". Basically, slots can be empty (E), can carry a cell for the node (FN), or a cell that cares to exit at output 1 (C1) or output 2 (C2), or a don't care (DC) cell.

3.2 Results/Improvement

The transmission results obtained for H-S&F are shown below:



(a) (b)

Figure 9: a) BER(h) performance of H-S&F versus transmission power, b) BER versus power coupling coefficient.

Figure 9(a) shows the conditional BER(h) for hop number 9 versus transmission power at g=1 for the channel at 1556 nm, the one with the worst gain. Results are shown for MS topology with a coupling coefficient of $\alpha = -25$ dB and a bit rate of 2.5 Gb/s. For low transmission power the predominant beat noise is signal-ASE that increases with bit rate and for high transmission powers the signal-crosstalk beat dominates, a noise that is bit rate independent.

Figure 9(b) shows BER results against α for a network of 64 nodes at g=1. It shows results for one buffer, hot potato, and H-S&F. Observe that H-S&F performs much better than one buffer and hot-potato deflection routing [12]. The reason is that H-S&F has the minimal average propagation delay and therefore the impact of intraband crosstalk and ASE noise is minimized. Note that the BER of H-S&F when all nodes transmit at R=2.5 Gb/s is below 10^{-9} for any α.

4 Promise of Wavelength Conversion

Wavelength translation (conversion) mitigates the blocking of cells substantially in cross-connected networks [15], [16]. By increasing the number of wavelengths and employing wavelength conversion the probability of deflection can be reduced which, in turn, leads to a significant imporvement in the teletraffic performance of the network.

Architecture of the wavelength translation (conversion) node is shown in Figure 10. All the submodules are interconnected and there is a control unit which decides transmission operations. Figure 11 shows the physical and logical submodule structures.

The main benefits of wavelength translation can be seen in Figure 12. By using more wavelengths, the average propagation delay experienced for a given throughput can be reduced substantially [14]. Different access schemes can be utilized for the node structure shown in Fig. 11(a). The main results are shown in Fig. 13.

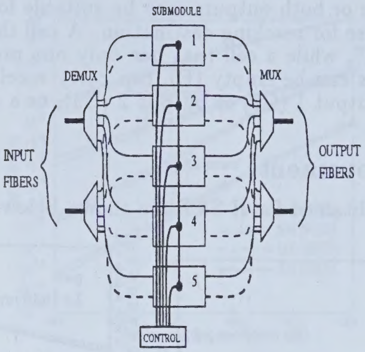
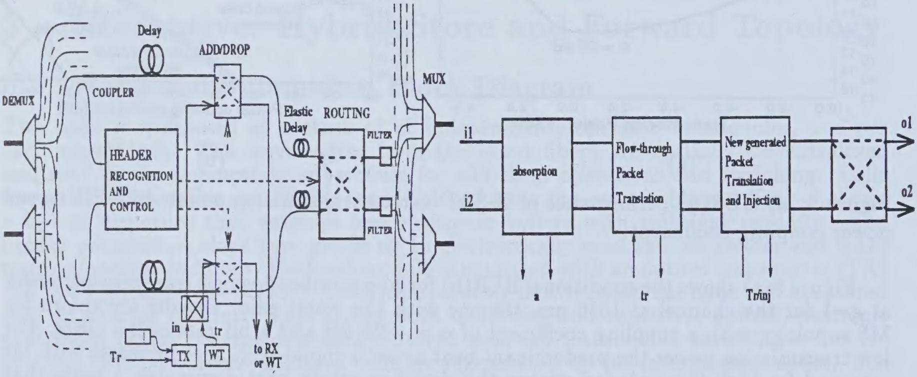

SUBMODULE
1
2
3
4
5
DEMUX
MUX
INPUT FIBERS
OUTPUT FIBERS
CONTROL

Figure 10: Architecture of the wavelength translation node. All the submodules are interconnected and there is a central control unit which decides transmission operations.


Delay
ADD/DROP
COUPLER
DEMUX
HEADER RECOGNITION AND CONTROL
Elastic Delay
ROUTING
FILTER
FILTER
MUX
in
tr
Tr
TX
WT
to RX or WT
i1
i2
absorption
Flow-through Packet Translation
New generated Packet Translation and Injection
o1
o2
a
tr
Tr/inj

(a) (b)

Figure 11: a) Physical and b) logical submodule structure.

Slotted hot-potato meshed networks with 64 nodes employing more than 4 wavelengths and wavelength conversion can work at full load with a hop delay of less than one hop from the zero-load hop delay [16]. The probability of deflection can be made quite low by increasing the number of wavelengths but as the load increases, the number of wavelengths required may become extremely large.

Results also indicate that, more than 3 wavelengths are needed for wavelength conversion routing to match the contention resolution capability of one delay-line per submodule [16]. However, optical buffers increase the accumulation of intra-band crosstalk and ASE noise, which may severely degrade the quality of received signals. Hence, only when possible it is preferrable to add a limited number of optical buffers per submodule in order to reduce the number of wavelength converters and, therefore, hardware cost. The main elements in favor of wavelength conversion are: 1)It can suppress the accumulation of noise such as ASE and crosstalk for OOK modulation; and 2) It doesn't add extra latency, this being a minor plus for high-speed packet switching.

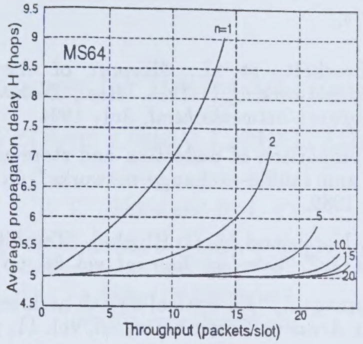
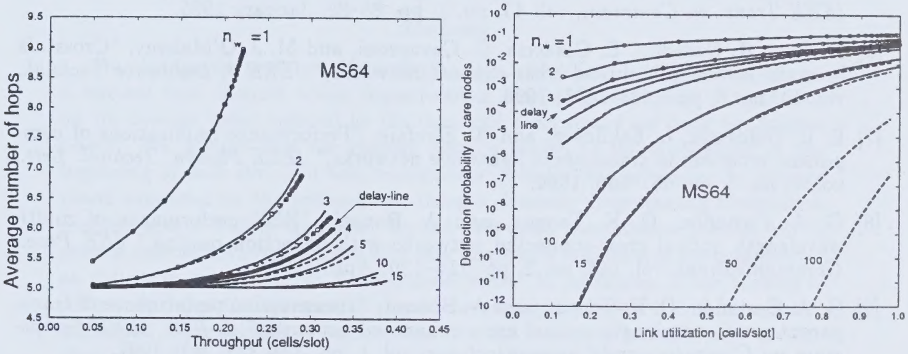


Figure 12: Average propagation delay versus throughput for different no. of wavelengths.



(a)

(b)

Figure 13: a) Delay versus throughput, b) Deflection probability versus link utilization.

5 Conclusions

Different routing strategies reviewed in this paper lead to the following conclusions:

- Contrary to conventional wisdom in networking research, transmission performance of all-optical networks, in addition to teletraffic performance, will be crucial in the big picture.
- Transmission performance is strongly affected by the node and device characteristics.
- Deflection Routing with no buffering in the node seems to have poor performance.
- Hybrid Store-and-Forward Topology has good potential if priority is given to buffered packets.
- Wavelength conversion is a promising new technology which, in conjunction with deflection routing, could substantially improve the Throughput-Delay characteristic of these networks.

References

[1] R. Blahut, C. A. Brackett, et al., "Report of the ARPA/NSF workshop on research in gigabit networking," *Tech. Rep., National Science Foundation, http://www.cise.nsf.gov/ncri/networks.html*, July 1994.

[2] N. F. Maxemchuk, "Comparison of deflection and store-and-forward techniques in the manhattan street and shuffle-exchange networks," in *Proc. IEEE INFOCOM '89*, pp. 800–809, Apr. 1989.

[3] A. S. Acampora, M. J. Karol, and M. G. Hluchyj, "Terabit lightwave networks: the multihop approach," *AT&T Technical Journal*, vol. 66, no. 6, pp. 21–34, Dec. 1987.

[4] I. Chlamtac and A. Fumagalli, "An optical switch architecture for manhattan networks," *IEEE J. Select Areas in Communication*, vol. 11, pp. 550–559, May 1993.

[5] F. Forghieri, A. Bononi, and P. Prucnal, "Analysis and comparison of hot-potato and single-buffer deflection routing in very high bit rate optical mesh networks," *IEEE Trans. on Commun.*, vol. 43, no. 1, pp. 88–98, January 1995.

[6] J. Zhou, R. Cadeddu, E. Casaccia, C. Cavazzoni, and M. J. O'Mahony, "Crosstalk in multi-wavelength optical cross-connect networks," *IEEE J. Lightwave Technol.*, vol. 14, no. 6, pp. 1423–1435, 1996.

[7] E. L. Goldstein, L. Eskildsen, and A. Elrefaie, "Performance implications of component crosstalk in transparent lightwave networks," *IEEE Photon. Technol. Lett.*, vol. 6, no. 5, pp. 657–660, 1994.

[8] G. A. Castañón, O. K. Tonguz, and A. Bononi, "BER performance of multi-wavelength optical cross-connected networks with deflection routing," *IEE Proc.-Communications*, vol. 144, no. 2, pp. 114–120, April 1997.

[9] G. A. Castañón, O. K. Tonguz, and A. Bononi, "Transmission performance of transparent multi-wavelength optical cross-connected networks," in *Proc. IEEE Symposium on Computers and Communications*, vol. 1, pp. 323–329, July 1997.

[10] C. R. Giles and E. Desurvire, "Modeling erbium-doped fiber amplifiers," *IEEE/OSA J. Lightwave Technol.*, vol. 9, no. 2, pp. 271–283, Feb. 1991.

[11] M. Kovacevic and A. Acampora, "On the benefits of wavelength translation in all-optical clear channel networks," *IEEE J. Select. Areas Commun.*, vol. 14, no. 5, pp. 868–880, June 1996.

[12] G. A. Castañón, O. K. Tonguz, and A. Bononi, "Combined teletraffic/transmission performance of multi-wavelength optical cross-connected networks using hybrid-store-and-forward," in *Proc. IEEE Global Telecommunications Conference (GLOBECOM'97), Sesion S24.2*, vol. 2, pp. 856–862, November 1997.

[13] S. Subramaniam, M. Azizoglu, and A. K. Somani, "All-optical networks with sparse wavelength conversion," *IEEE/ACM Trans. Networking*, vol. 4, pp. 544–557, August 1996.

[14] G. A. Castañón, O. K. Tonguz, and A. Bononi, "On the benefits of wavelength translation in datagram all-optical networks," *IEE Electronics Letters*, vol. 33, no. 18, pp. 1567–1568, August 1997.

[15] T. Durhuus, B. Mikkelsen, C. Joergensen, S. L. Danielsen, and K. E. Stubkjaer, "All-optical wavelength conversion by semiconductor optical amplifiers," *IEEE J. Lightwave Technol.*, vol. 14, no. 6, pp. 942–954, June 1996.

[16] A. Bononi, G. Castañón, and O. K. Tonguz, "Analysis of hot-potato optical networks with wavelength converters," *IEEE/OSA J. of Lightwave Technology*, vol. 17, no. 4, pp. 525-534, April 1999.

Transmission Policies for Time Varying Channels with Average Delay Constraints*

B. E. Collins and R. L. Cruz

Dept. of Electrical & Computer Engineering
University of California, San Diego
La Jolla, CA 92093-0407
[becollin,rcruz]@ucsd.edu

Abstract

We consider the problem of transferring data with a fixed level of reliability over a discrete time channel whose characteristics vary with time, with a constraint on the average delay suffered by the data and a constraint on peak transmitter power. We assume that the channel state is known to the transmitter at the beginning of each slot, and seek transmission policies that minimize the average power expended by the transmitter. Using a dynamic programming formulation coupled with a duality argument, we find that good transmission policies will in general take action in accordance with how much data is queued at the transmitter, as well as in accordance with the channel state. In particular, if the backlog is sufficiently large the transmitter should use the channel even if it is in a "bad" state, and if the backlog is sufficiently small the transmitter should only use the channel when it is in a "good" state. However, we find that the time scale of the variations in the channel relative the maximum tolerable average delay is a crucial factor influencing the qualitative features of the optimal transmission policy.

1 Introduction

The primary objectives of the design of present day cellular networks have been to carry voice traffic. Voice traffic has narrowly defined rate, delay and reliability requirements and the current second generation cellular systems appear to satisfy these requirements quite well. However, as cellular technology evolves to accommodate the mix of traffic classes envisioned in an integrated service cellular network some of the techniques currently used to combat the time-varying nature of the wireless link must be re-examined. For example, it was shown in [9] that the common technique of interleaving to distribute bursts of errors actually reduces the throughput performance of a packet data stream when ARQ is employed for error correction at a higher network layer.

The various types of packet data currently encountered in the wired network present a wide range of error and delay tolerance. Delay jitter, which is a significant impairment to voice traffic, can be easily tolerated by many types of packet data traffic. However,

---

*This research was supported in part by the National Science Foundation under grant NCR-97-25568, and by industry members of the UCSD Center for Wireless Communications, with matching funds under the State of California Communications Research Program (CoRe).

average delay is still an important consideration for some types of packet data traffic. Here we are concerned with average delay constrained streams and endeavor to determine how to best transmit such a stream in order to meet the delay constaint and minimize the transmit energy expended.

In this paper, we consider a single stream of data to be transmitted across a link whose characteristics change with time due to fading, where the average delay suffered by the data at the transmitter must be no more than a prescribed maximum average delay, $\bar{D}_{max}$, and where the transmitter is governed by a peak power constraint. We use a discrete time model where time is divided into equal length slots, and assume that the magnitude of the fading is constant over each slot. Furthermore, we assume that the transmitter has knowledge of the fading state at the beginning of each slot. The data stream must be transmitted across the link with a fixed reliability. Thus, the maximum amount of data that can be transmitted during a slot decreases with the amount of fading that exists during the slot. For example, the information rate of an error correcting code could be be adapted to the state of the channel. For simplicity, we assume the fading process is a two-state Markov chain, corresponding to the Gilbert-Elliott channel model [3, 2].

We seek transmission policies that minimize the average power expended in each slot. Using a dynamic programming formulation coupled with a duality argument, we are led to consider "*critical backlog*" transmission policies. In such policies, there is a critical backlog value associated with each fading state. If the backlog of data at the beginning of a slot is less than the critical backlog value associated with the current fading state, then no data is sent during the slot. We say that the transmitter is "stingy" in this case. On the other hand, if the backlog of data at the beginning of the slot is greater than the critical backlog value associated with the current fading state, then the amount of data sent during the slot decreases the backlog at the end of the slot down to the critical backlog value, unless this is not possible due to the peak power constraint; In this latter case, as much data is sent as is possible without violating the peak power constraint, and we say that the transmitter is "desperate." We perform a brief numerical study and find that critical backlog policies appear to be near optimal.

Thus, we claim that a good transmission policy should use the current backlog, as well as the current channel state, to determine the action to be taken in each slot. In the "good" channel state, the transmitter should always send as much data as possible. In the "bad" channel state, roughly speaking, the transmitter should be stingy if the current backlog is "small" and wait for the channel state to get good; If the current backlog is "large" then the transmitter should be desperate and transmit as much data as possible, even if the channel state is bad.

In general, there are three regimes that arise, which are differentiated by the relative magnitudes of $\bar{D}_{max}$ and the expected holding times of the fading states. If the expected holding times are much greater than $\bar{D}_{max}$, then the optimal critical backlog value is zero, corresponding roughly to desperation independent of the current backlog. At the other extreme, if the expected holding times are much less than $\bar{D}_{max}$, then the optimal critical backlog value is infinite, corresponding to stinginess independent of the current backlog. In between these extremes, the critical backlog value is finite and the transmission policy should depend on the current backlog. We believe there may be many scenarios of interest, e.g. transfer of e-mail, where transmission policies corresponding to this latter regime may have practical application.

Some of the existing literature on the topic of *power control* is relevant to our study.

Goldsmith and Varaiya [4] showed that the optimal power control scheme for a single-user, Rayleigh fading channel with a power constraint is *waterfilling* in time. This scheme calls for transmitting at a power level equal to the constraining power level minus the noise power level. Knopp and Humblet [5] extended these results to show that the sum of rates capacity of a multiuser, frequency-flat fading channel is achieved using a power control scheme which only allows the user with the best channel to transmit at any given time. Both of these works consider *Shannon* capacity which is typically only achievable with long encoding/decoding delays. Hanly and Tse [6] point out that Shannon capacity is not even approximately achievable for delay sensitive data. These authors propose an alternate definition for capacity for delay sensitive data. Named *delay-limited* capacity, this capacity is defined as the maximum rate that can be achieved with delays *independent* of the correlation structure of the fading. This definition appears to be appropriate for delay sensitive applications which require a guaranteed *instantaneous* transmission rate. However, this definition of channel capacity appears to be pessimistic for delay sensitive streams which can tolerate significant delay jitter. In this study we are concerned with such streams with only an average delay constraint. Finally, we note that Wilhelmsson and Milstein [8] have considered the use of block codes and interleavers over Gilbert-Elliott channels with a delay constraint, and found that short codes sometimes offer better performance when the interleaver depth must be kept short due to the delay constraint.

2 Model

We now describe the model of the system under study in more detail. Time is divided into equal length intervals called slots, and the slots are indexed by the integers. Data arrives randomly to a transmitter to be transferred to a receiver across a time varying channel. We define a_n to be a random variable that represents the amount of data that arrives to the transmitter during slot n. This data can be transmitted in slot $n+1$ at the earliest, and may be transmitted in an even later slot. Thus, the transmitter has the capability to queue data. We define u_n to be the amount of data transmitted in slot n. In addition, we define x_n to be the amount of data queued at the transmitter at the beginning of slot n, also called the *backlog* at slot n. We require that $u_n \leq x_n$. The backlog process $\{x_n\}$ therefore satisfies the recursion

$$x_{n+1} = x_n - u_n + a_n \ . \tag{1}$$

We shall assume that the arrival process $\{a_n\}$ is a sequence of independent and identically distributed random variables. We let $\lambda = E[a_n]$.

The channel can be in one of two states, labeled G (short for "good") and B (short for "bad"). The state of the channel during slot n is known to the transmitter at the beginning of slot n, and is denoted by s_n. Thus, $s_n = G$ (resp. $s_n = B$) if the channel is in the good (resp. bad) state during slot n. The signal is attenuated more in the bad state than in the good state. A reliability requirement dictates that the data must be transferred across the channel with a constant signal to noise ratio. In particular, the energy spent by the transmitter during slot n is given by $w_n u_n$, where $w_n = w^G$ if $s_n = G$ and $w_n = w^B$ if $s_n = B$, and where $w^G < w^B$. We impose a peak power constraint on the transmitter. Specifically the transmitter can expend at most P_{max} units of energy during any slot (we normalize time so that P_{max} is also the peak power allowed). Combining these facts, we see that the amount of data transmitted during slot n must satisfy the constraint

$$0 \le u_n \le \min\{x_n, P_{max}/w_n\} \ . \tag{2}$$

We shall require that u_n be a deterministic function of x_n and s_n, i.e. $u_n = u_n(x_n, s_n)$, and shall generically denote a sequence of such functions as a policy π.

We assume that $\{s_n\}$ is a Markov process that is independent of the arrival process, and use the notation $q_{XY} = P(s_{n+1} = Y | s_n = X)$. Thus, for example, the number of slots that the channel stays in the bad state between state transitions is geometrically distributed with mean $1/q_{BG}$.

We require that the average delay suffered by data at the transmitter be no more than $\bar{D}_{max}$. Assuming that the backlog process $\{x_n\}$ is stationary, by Little's result the average delay is given by $\bar{x}/\lambda$, where $\bar{x} = E[x_n]$. Equivalently, we require that $\bar{x} \le \lambda \bar{D}_{max}$.

We approach the problem by considering it as an optimal control problem over a finite time horizon, say over slots $0, 1, 2, \ldots N-1$. In order to reflect the average delay constraint, we consider policies π such that

$$E[\sum_{n=1}^{N} x_n] \le \lambda \bar{D}_{max} N \ . \tag{3}$$

Let $\mathcal{B}$ be the set of all such policies.

We thus seek to minimize the total energy expended over the time horizon over the set of all policies π belonging to $\mathcal{B}$, i.e. we consider the following optimization problem:

$$\min_{\pi} \left\{ E[\sum_{n=0}^{N-1} w_n u_n] \right\} \text{ subject to } \pi \in \mathcal{B} \ . \tag{4}$$

3 Analysis

Let $f(\bar{D}_{max})$ be the optimal value of the objective function in our problem, i.e.

$$f(\bar{D}_{max}) = \min_{\pi \in \mathcal{B}} \left\{ E[\sum_{n=0}^{N-1} w_n u_n] \right\} \ . \tag{5}$$

Since direct evaluation of $f(\bar{D}_{max})$ appears difficult, we will use duality (which is also related to the so-called "penalty function" approach). In particular, defining β as a "dual variable," we define the dual function

$$g(\beta) = \min_{\pi} \left\{ E[\sum_{n=0}^{N-1} w_n u_n + \beta \sum_{n=1}^{N} x_n] \right\} \ . \tag{6}$$

Assuming we can calculate $g(\beta)$, we can then lower bound $f(\bar{D}_{max})$ by using the convex conjugate of g, i.e. for $\beta \ge 0$ we have

$$\begin{aligned} f(\bar{D}_{max}) &= \min_{\pi \in \mathcal{B}} \left\{ E[\sum_{n=0}^{N-1} w_n u_n] \right\} \\ &\ge \min_{\pi \in \mathcal{B}} \left\{ E[\sum_{n=0}^{N-1} w_n u_n] + \beta \left(E[\sum_{n=1}^{N} x_n] - \lambda \bar{D}_{max} N \right) \right\} \\ &\ge \min_{\pi} \left\{ E[\sum_{n=0}^{N-1} w_n u_n] + \beta \left(E[\sum_{n=1}^{N} x_n] - \lambda \bar{D}_{max} N \right) \right\} \\ &= g(\beta) - \beta(\lambda \bar{D}_{max} N) \ , \end{aligned}$$

so that

$$f(\bar{D}_{max}) \geq \sup_{\beta \geq 0} \{g(\beta) - \beta(\lambda \bar{D}_{max} N)\} . \tag{7}$$

Note that this lower bound will be tight if f is a convex function of $\bar{D}_{max}$.

The problem of calculating $g(\beta)$ is relatively straightforward, using the approach of dynamic programming [1]. In particular, define

$$J^N(x,s) = \min_{\pi} E[\sum_{n=0}^{N-1} w_n u_n + \beta \sum_{n=1}^{N} x_n \mid x_0 = x, s_0 = s] . \tag{8}$$

Then $J^N(x,s)$ satisfies the recursions

$$J^{N+1}(x,G) = \min_{u:0 \leq u \leq \min\{x, P_{max}/w^G\}} \{\beta(\lambda + x) + (w^G - \beta)u + q_{GG}E[J^N(x - u + a_0, G)] + q_{GB}E[J^N(x - u + a_0, B)]\} \tag{9}$$

and

$$J^{N+1}(x,B) = \min_{u:0 \leq u \leq \min\{x, P_{max}/w^B\}} \{\beta(\lambda + x) + (w^B - \beta)u + q_{BG}E[J^N(x - u + a_0, G)] + q_{BB}E[J^N(x - u + a_0, B)]\} . \tag{10}$$

The initial conditions of this recursion are

$$J^1(x,G) = \min_{u:0 \leq u \leq \min\{x, P_{max}/w^G\}} \{\beta(\lambda + x) + (w^G - \beta)u\} \tag{11}$$

and

$$J^1(x,B) = \min_{u:0 \leq u \leq \min\{x, P_{max}/w^B\}} \{\beta(\lambda + x) + (w^B - \beta)u\} . \tag{12}$$

Note that $J^1(x,G)$ and $J^1(x,B)$ are convex functions of x. It can be shown by induction using (9) and (10) that in fact $J^N(x,G)$ and $J^N(x,B)$ are convex functions of x for all N. From this it follows that u^*, the optimizing value of u in (9) and (10), takes on the form

$$u^* = u^*(x,s) = \begin{cases} 0 & , \text{if } x \leq x_s^* \\ x - x_s^* & , \text{if } x_s^* \leq x \leq x_s^* + P_{max}/w^s \\ P_{max}/w^s & , \text{if } x > x_s^* + P_{max}/w^s , \end{cases} \tag{13}$$

for some number x_s^*, which we call a critical backlog value. These are the critical backlog type policies discussed in the introduction. We can easily augment the dynamic programming recursions in order to calculate

$$E[\sum_{n=1}^{N} x_n \mid x_0 = x, s_0 = s] \tag{14}$$

and

$$E[\sum_{n=0}^{N-1} w_n u_n \mid x_0 = x, s_0 = s] \tag{15}$$

for the corresponding optimal policy π, for each value of β. This can be used to generate upper bounds for $f(\bar{D}_{max})$. In our numerical experiments, we have found these upper and lower bounds for $f(\bar{D}_{max})$ to be close, suggesting that critical backlog type policies are near optimal.

4 Numerical Analysis

We performed a series of numerical calculations to evaluate the minimum average energy required per slot as a function of the maximum allowed average delay $\bar{D}_{max}$. In order to realize a finite state space, we constrained a_0, as well as u_n, to take on only integer values. In addition, we truncated the state space to a finite maximum backlog, x_{max}.

In particular, a computer program was developed to implement the dynamic programming algorithm [1] for computing $g(\beta)$ over a range of values of β given a Bernoulli arrival distribution, the Gilbert-Elliott channel transition probabilities and per packet transmission costs in the good and bad states and the peak power constraint. While only finite horizon, finite state solutions can ever be exactly computed we apply the *value iteration* and *approximating sequence* techniques described in [7] to compute good approximations to the optimal policies and corresponding *average cost* for the infinite horizon, denumerably infinite state case. In other words, using these techniques we were able to compute policies and corresponding average transmit energies that are stable as $N \to \infty$ *and* as the size of the backlog state space, x_{max}, becomes large. The results of this computation can then be used to select the optimal policy that satisfies the average delay constraint or to study the behavior of the optimal policy over a range of conditions (e.g., channel fading and arrival rate).

Defining

$$\delta^N(x,s) = J^N(x,s) - J^{N-1}(x,s)$$

we terminate iterative calculation of the N-horizon optimum solutions when

$$\max_{x,s}[\delta^N(x,s) - \delta^{N-1}(x,s)] \leq HorizonTolerance \tag{16}$$

The value of HorizonTolerance was selected experimentally to be sufficiently small that when (16) is satisfied then the computed policy is stable and unchanging with further iterations.

This computational approach is described in [7] where it is referred to as the *Value Iteration Algorithm.* This algorithm is proven to converge to an optimal stationary policy for the *infinite horizon* (but finite state) average cost optimization criteria under some assumptions which our problem formulation satisfy. Furthermore, the value of δ is the minimum average cost (i.e., the minimum average transmit energy per timeslot).

Both the upper and lower bounds for $f(\bar{D}_{max})$ for one particular scenario are plotted in Figure 1 (arrivals uniformly distributed over [0,14], $w^G = 10$, $w^B = 100$, $q_{GB} = q_{BG} = 0.16667$, $P_{max} = 200$, HorizonTolerance = 10^{-7}, $x_{max} = 700$). The range of average delay in this plot is limited to illustrate the difference between the two curves. Notice that the lower bound is quite tight. In fact the lower and upper bounds are equal for some values of average delay. The points where these bounds touch occur at values of average delay that are achievable by an optimal policy which only transmits an integer number of packets per slot.

We investigated several scenarios to determine the optimal policies for transmitting packet data with average delay constraints and how these policies depend on channel memory and arrival rate. Some of the more interesting results are summarized in the remainder of this section.

Figure 2 shows how the minimum average transmission cost depends on the tolerable average delay and on the channel memory (arrivals uniformly distributed over [0,12], $w^G = 10$, $w^B = 100$, $P_{max} = 200$, HorizonTolerance = 10^{-7}, $x_{max} = 700$). To simplify our discussion, we chose $q_{BG} = q_{GB}$ which implies that the good and bad states of the channel are equally probable. We defined the "channel memory" to be the average holding time $1/q_{BG}$. Notice that increasing channel memory does not result in an increase in the required average transmit energy as long as a higher average delay is tolerable. It is not surprising that the required shift in the tolerable average delay is roughly the same as the increase in channel memory.

Figure 3 shows how the critical backlog value in the computed optimal policy depends on the average delay constraint and on the channel memory (arrivals uniformly distributed over [0,12], $w^G = 10$, $w^B = 100$, $q_{BG} = q_{GB} = \{0.125, 0.10, 0.0833\}$, $P_{max} = 200$, HorizonTolerance $= 10^{-7}$, $x_{max} = 700$). Notice that for each value of channel memory we can divide the horizontal axes (i.e., tolerable average delay) into three regions. The leftmost region extends from an average delay value of zero up to (but not including) the last point where the policy critical backlog value equals 1. This region contains the range of average delay values that are *not* achievable. In this region of tolerable average delay values the optimal thing to do is transmit the maximum amount of energy allowed in every timeslot (or the entire backlog for small values of backlog) regardless of the channel state. Recall we have described the optimal policy in this region *desperate.* The rightmost region starts at the point where $x^* = x_{max}$ and includes all larger values of tolerable average delay. In this region the optimal policy is *stingy.* That is, the optimal thing to do is only transmit when the channel is in the good state. In the region between these two extremes the critical backlog value in the optimal policy increases monotonically with increasing tolerable average delay.

Figure 4 shows how the critical backlog value depends on the average delay constraint and on the packet arrival rate (arrivals uniformly distributed over {[0,12],[0,14],[0,16]}, $w^G = 10$, $w^B = 100$, $q_{BG} = q_{GB} = 0.16667$, $P_{max} = 200$, HorizonTolerance $= 10^{-7}$, $x_{max} = 700$). It is not surprising that increasing arrival rate has a very pronounced effect on the critical value.

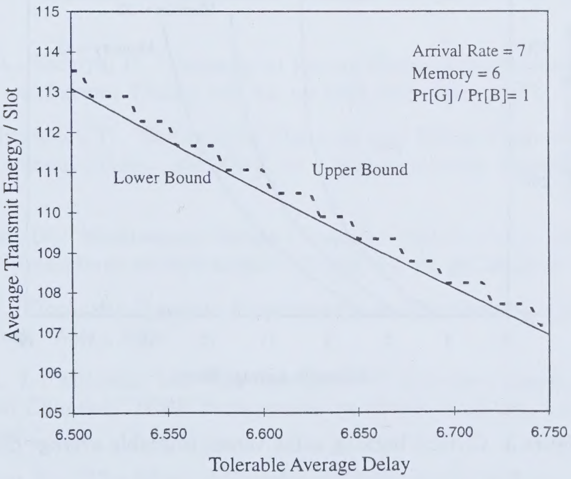


Figure 1: Upper and Lower Bounds for Energy-Delay Curve.

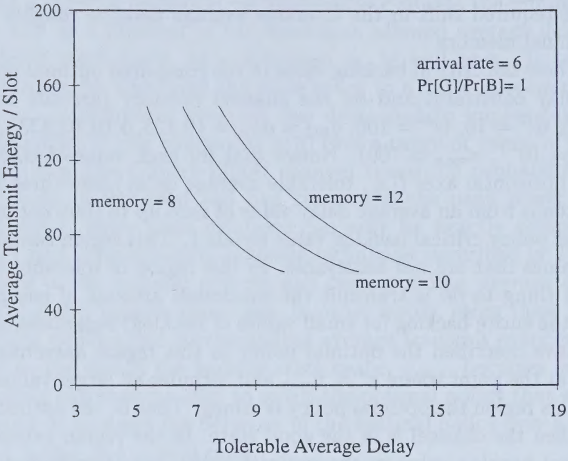


Figure 2: Energy-Delay curves for various channel memory values.

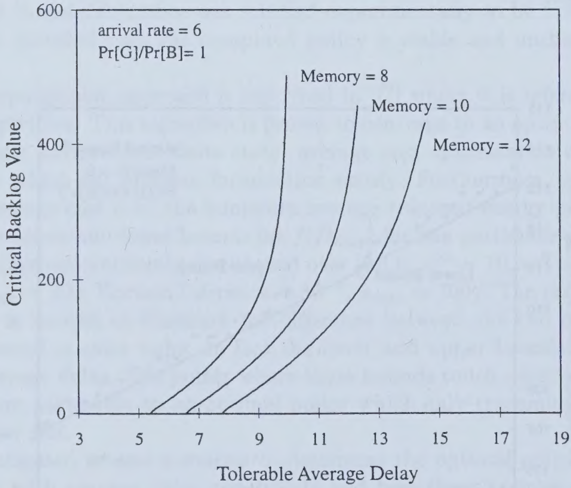


Figure 3: Critical backlog value versus tolerable average delay.

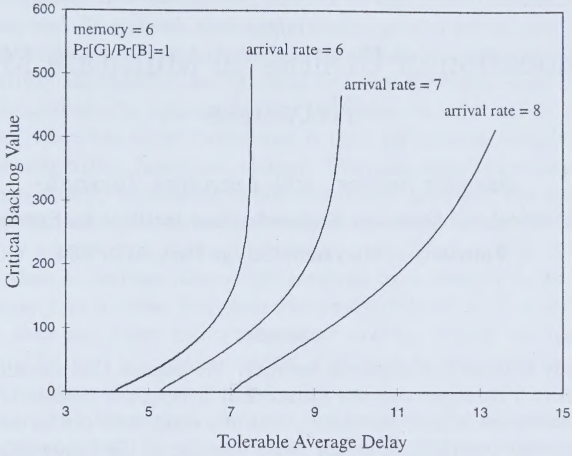


Figure 4: Dependence of critical backlog value on arrival rate.

References

[1] Bertsekas, D. P., *Dynamic Programming: Deterministic and Stochastic Models*, Englewood Cliffs, NJ: Prentice-Hall, 1987.

[2] Elliott, E. O., "Estimates of Error Rates for Codes on Burst-Noise Channels", *BSTJ*, Vol. 42, pp.1977-1997, September 1963.

[3] Gilbert, E. N., "Capacity of a Burst-noise Channel", *BSTJ*, Vol. 39, pp. 1253-1265, 1960.

[4] Goldsmith, A., Varaiya, P., "Capacity of Fading Channels with Side Information," *IEEE Trans. on Inform. Theory*, vol. 43, pp.1986-1992, Nov. 1997.

[5] Knopp, R., Humblet, P., "Information Capacity and Power Control in Single-Cell Multiuser Communications," *Int. Conf. on Communications*, Seattle, Wash., June 1995.

[6] Hanly, S., Tse, D., "Multi-access Fading Channels: Part II: Delay-Limited Capacities," *IEEE Transactions on Information Theory*, vol. 44, pp. 2816-2831, Nov. 1998.

[7] Sennott, L. I., *Stochastic Dynamic Programming and the Control of Queueing Systems*, New York: Wiley, 1999.

[8] Wilhelmsson, L., Milstein, M., "On the Effect of Imperfect Interleaving for the Gilbert-Elliott Channel," *IEEE Transactions on Communications*, vol. 47, pp. 681-688, May, 1999.

[9] Zorzi, M., Rao, R., "The Effect of Correlated Errors on the Performance of TCP," *IEEE Communications Letters*, vol. 1, no. 5, pp. 127-129, September, 1997.

Fair Allocation of Utilities in Multirate Multicast Networks

Saswati Sarkar and *Leandros Tassiulas*
Dept. of Electrical and Computer Engineering and Institute for Systems Research
University of Maryland, College Park, MD, USA

Abstract

We study fairness in a multicast network. We assume that the source hierarchically encodes its signal and the hierarchical structure is predetermined. We study fair allocation of utilities, where utility of a bandwidth can be the number of layers or the bandwidth itself, or any other function of the bandwidth depending on system requirements. The utility function is not strictly increasing in general. Fairness issues become vastly different in this case as opposed to that when the utility function is strictly increasing. Computation of lexicographically optimal utility allocation becomes NP-hard in this case, while lexicographically optimal utility allocation is polynomial complexity computable when the utility function is strictly increasing. Furthermore, maxmin fair utility allocation may not exist in the general case. We introduce a new notion of fairness, maximal fairness. We propose a polynomial complexity algorithm for computation of maximally fair utility allocation. Even though, maximal fairness is a weaker notion of fairness, it coincides with lexicographic optimality and maxmin fairness, when maxmin fair utility allocation exists. So the algorithm for computing maximally utility allocation computes maxmin fair utility allocation, when the latter exists.

1 Introduction

Multicasting poses some specific fairness challenges. The fairness objective is that every receiver receives service at a rate commensurate with its capabilities and the capacity of the path leading to it from the source. A single rate of transmission per session is likely to either overwhelm the slow receivers or starve the fast ones, in absence of additional provisions. Multirate transmission should be used to counter network heterogenity. Multirate transmission can be attained using a hierarchical or a layered transmission scheme. In this approach, a signal is encoded into a number of layers that can be incrementally combined to provide progressive refinement. The receivers adapt to congestion by adding and dropping layers.

We study fair allocation of utilities, where utility of a bandwidth can be the number of layers, the bandwidth itself or any other function of the bandwidth, e.g., perceptual quality. Maxmin fairness[1] is a well accepted notion of fairness. A utility allocation is maxmin fair, if no receiver can be allocated a higher utility without hurting another receiver having equal or lower utility. A maxmin fair utility allocation may not exist if the utility functions are not strictly increasing. However, a maxmin fair utility allocation always exists for strictly increasing utility functions. Lexicographic optimality is another notion of fairness. A lexicographically optimal utility vector is one which maximizes its minimum component in a feasible set, subject to this maximization, it maximizes the second minimum, etc[1]. In general, lexicographically optimal utility allocation exists even when the utility function is not strictly increasing. However, computation of lexicographically optimal utility allocation is an NP-hard problem for non-strictly increasing utility functions. Lexicographically optimal utility allocation is identical to the maxmin fair utility allocation, when the latter exists and is thus polynomial complexity computable for strictly increasing utility functions, though. We can compute a maximally fair utility allocation for non strictly increasing utility functions, instead. We introduce the concept of maximal fairness more formally later, but a utility allocation is maximally fair if no other utility allocation is "fairer" in some sense. That is, if a utility allocation is maximally fair, then to increase the utility received by a receiver s, we must lower that of another receiver j to a value less than the new utility of s. If a utility allocation is maximally fair, then any other utility allocation will be "unfair" or "less fair" to some receiver. Maximal fairness is a weaker notion of fairness as compared to maxmin fairness and lexicographic optimality. But, maximal fairness has various desirable fairness properties,e.g., it coincides with maxmin fairness and lexicographic optimality when maxmin fair utility allocation exists. We discuss other desirable fairness properties of maximally fair allocations later. In a nutshell, maximal fairness is probably the best we can achieve for non-strictly increasing utility functions, in view of the nonexistence of maxmin fair utility allocation and computational complexity of lexicographically optimal utility allocation. We will present a polynomial complexity algorithm for computing maximally fair utility allocation in this paper. This algorithm yields a maxmin fair utility vector, if it exists. Our algorithm for computation of maximally fair allocation does not assume any properties specific to internet or ATM. So it is applicable in a very general scenario. Keeping in mind ATM networks, we have incorporated minimum utility requirements and maximum utility constraints in our model.

We review related work in this area. Well known network protocols for layered transmission, RLM (Receiver-driven Layered Multicast)[8] and LVMR (Layered Video Multicast with Retransmissions)[5] do not handle fairness among sessions very well, when there are multiple sessions competing for bandwidth[6]. An alternative layer allocation scheme has been proposed in [6]. There is empirical evidence that this scheme improves fairness among sessions for networks with multiple video sessions sharing only one link. But, there is no experimental or analytical evidence that the scheme works well for more complex networks, with sessions sharing several links with each other. Besides [6] does not compute actual rates or the number of layers allocated to the receivers in an arbitrary network, under some well defined notion of fairness. An algorithm for computation of maxmin fair rates, when hierarchical signal structure is completely flexible, has been proposed in [10]. Recently, [11] addressed the scenario with hierarchical signal structure predetermined. But, it considers the special case in which bandwidth consumed by every layer is the same irrespective of the particular layer or the source. It points out that maxmin fair layer allocation may not exist and computation of lexicographically optimal layer allocation is NP-hard even in this special case. It also presents an algorithm for computation of maximally fair layer allocations. However this algorithm does not yield a maximally fair layer allocation, in a more general case i.e., if different layers consume different bandwidths, or if the network can allocate any rate in a particular bandwidth range and only a discrete set of bandwidths in another range. The situation becomes significantly different in this general case. The general case is relevant in the current day networking scenario. For example, every additional layer induces an exponential increase in the total bit rate in the codec of [7]. A study of fair allocation of utilities will capture this general scenario. Incidentally, [3] presents an algorithm for computation of maxmin fair utilities. However the algorithm in [3] has been presented for unicast network only, whereas we consider multicast network. More importantly, the algorithm attains maxmin fair utilities for strictly increasing utility functions only. The operation of this algorithm is ambiguous for nonstrictly increasing utility functions and does not even generate a maximally fair utility allocation in general[11]. Again there are many examples of practical importance in which the utility functions do not increase strictly.

---

[1] Maxmin fairness and lexicographic optimality will be defined formally later.

2 Allocation of Utilities

We first motivate the study of fair allocation of utilities. The hierarchical signal structure is often predetermined and different for different sources. For example, a network can have two sessions and a single link. Source of the first session transmits unit bandwidth layers. Every layer of the second session consumes 2 units of bandwidth. Let the capacity of the link be 9 units. A bandwidth allocation allocating 5 units of bandwidth to the first session and 4 units to the second one is intuitively fair (predetermined layer bandwidths do not allow allocation of 4.5 units each). However, this allocation gives session 1, 5 layers and session 2, 2 layers. Intuitively, $(3, 3)$ is a "fairer" layer allocation. The corresponding bandwidth allocation, $(3, 6)$ does not seem fair. So, fair allocation of layers and bandwidths are different objectives. We think the fairness objective should depend on the specific requirements of particular networks. So we study fair allocation of utilities, where utility of a bandwidth can be the number of layers, the bandwidth itself or any other function of the bandwidth, e.g., perceptual quality.

There can be various possibilities in hierarchical coding. The hierarchical structure may be completely flexible, i.e., given the desired rates of various receivers, the source generates as many layers as distinct receiver rates and tunes the layer bandwidths so as to match the receiver rates. The hierarchical structure may be predetermined, i.e., layer bandwidths can not be tuned to match any desired receiver rates and receivers can not partially subscribe to layers. Also, it is possible that layer bandwidths can be fine tuned in certain ranges, while the granularity is coarse in other ranges. Hierarchical structure of different sources may be different, i.e., signal structure of some sessions may be flexible, while the signal structure may be predetermined for some others. Bandwidths consumed by layers may be different for different sources. Bandwidths consumed by different layers of the same source may also be different. We model all these scenarios by choosing the utility functions appropriately. We discuss the choice of utility functions later, but in general, the utility functions will not be strictly increasing. Fairness under this assumption is vastly different from the case in which the utility functions are strictly increasing.

Formally, "utility" is a function of the bandwidth. We assume that the utility function for receivers of the same session is the same. "Utility set" of session i, A_i is the range of the utility function of session i. Receivers of a session i can possibly receive a utility if it belongs to utility set of session i, A_i. We assume that the utility function satisfies the following technical assumptions for every i.

1. $A_i \cap (x, \infty) \neq \phi, \forall x \in R$.

2. Given any $y \geq 0$, $\max(A_i \cap [0, y])$ exists.
3. If $x \notin A_i$, then $\min(A_i \cap (x, \infty))$ exists.
4. Given any utility x, there always exists a minimum bandwidth required for session i to attain utility x, i.e., the set $\{y : u_i(y) = x\}$ has a minimum, where $u_i(y)$ is the session i utility corresponding to bandwidth y. The minimum bandwidth for utility x is denoted $\Upsilon_i(x)$.

The various scenarios described before, e.g., hierarchical structure flexible, partly flexible and partly predetermined, completely predetermined, etc. can be modelled by choosing the utility set suitably for every session i. We describe a few possible choices in Figure 1(a). The first assumption indicates that the utility set is not upper bounded. If necessary, we can impose maximum utility constraints separately. The other assumptions are more technical. Assumption (2) indicates that if there exists elements in a utility set arbitrarily close to but less than a positive real number z, then z is an allowable utility. Assumption (3) indicates that, if a real number x is not an allowable utility then the set of utilities greater than x has a minimum. The last three assumptions exclude some functions from being valid utility functions. Refer to Figure 1(b) for details.

Observe that the utility function is non strictly increasing in many cases(Figure 1(a)). Any utility function satisfying properties (1) to (4) is acceptable. The assumptions are general in nature and hence a large choice of utility functions are acceptable. Also the utility function of receivers of the same session must be the same. This assumption is reasonable, because utility structure, (e.g., hierarchical structure) often depends on source and there is only one source per session.

3 Network Model

We consider an arbitrary topology network with N multicast sessions. A multicast session is identified by the pair (v, U), where v is the source node of the session and U is the group of intended destination nodes. We assume that the traffic from node v is transported across a predefined multicast tree to nodes in U. The tree can be established during connection establishment phase if the network is connection oriented or can be established by some well known multicast routing protocol like DVMRP[4], CBT[2], etc. in an internet type network.

We call every source destination pair of a session a virtual session. For example, if a session n has source v_n and destination set U_n, where $U_n = \{u_{n1}, \ldots, u_{nt}\}$, then this session would correspond to t virtual sessions, $(v_n, u_{n1}), \ldots, (v_n, u_{nt})$. To ensure fairness in a multirate network, we need to consider fair utility allocation for the virtual sessions separately, instead of those for the overall sessions.

An $M-$ dimensional vector $\vec{r}$ is a feasible utility allocation if

1. $r_s \in A_{\chi(s)}$, $\forall\ s \in \{1, \ldots, M\}$, where $\chi(s)$ is the session of virtual session s.
2. $\mu_i \leq r_i \leq p_i$ $\forall i$, where μ_i and p_i are respectively the minimum and maximum utiliities of virtual session i, $p_i \geq \mu_i \geq 0$,
3. Total bandwidth consumed in every link is less than or equal to the capacity of the link. Bandwidth consumed in a link is the sum of the bandwidths consumed by different sessions traversing the link. Bandwidth consumed by a session in a link

is the bandwidth consumed by the *session link utility*, where session link utility is the maximum utility allocated to virtual sessions of the same session traversing the link. Let $n(l)$ denote the set of sessions traversing link l, $m(k,l)$ denote the set of virtual sessions of session k passing through link l, C_l denote the capacity of link l, λ_{il} denote the session link utility of session i in link l and $\Upsilon_i(x)$ denote the minimum bandwidth required for session i to attain utility x . Then $\lambda_{il} = \max_{j \in m(i,l)} r_j$ and

$$\sum_{i \in n(l)} \Upsilon_i(\lambda_{il}) \leq C_l \quad \text{(capacity condition).}$$

A feasible utility allocation is maxmin fair if it is not possible to maintain feasibility and increase the utility of a virtual session without decreasing that of any other virtual session which has equal or lower utility. More formally, a feasible utility allocation vector $\vec{r}^1$ is maxmin fair if it satisfies the following property with respect to any other feasible utility allocation vector $\vec{r}^2$: if there exists i such that the ith component of $\vec{r}^2$ is strictly greater than that of $\vec{r}^1$ ($r_i^2 > r_i^1$), then there exists j such that the jth component of $\vec{r}^1$, r_j^1 is less than or equal to the ith component of $\vec{r}^1$, r_i^1 ($r_j^1 \leq r_i^1$) and the jth component of $\vec{r}^2$ (r_j^2) is strictly less than the jth component of $\vec{r}^1$ ($r_j^2 < r_j^1$). A maxmin fair utility allocation may not exist in general for non strictly increasing utility functions.

Given a $M-$ dimensional vector $\vec{V}$, define its lexicographically ordered version $\hat{V}$ as follows: $\hat{v}_j = v_k$ for some k and $\hat{v}_1 \leq \hat{v}_2 \leq \ldots \hat{v}_M$. In other words, components of $\hat{V}$ are an ordered version of those of $\vec{V}$. A vector $\vec{r}^1$ is lexicographically greater than another $\vec{r}^2$ if there exists i such that $\hat{r}_i^1 > \hat{r}_i^2$ and $\hat{r}_j^1 = \hat{r}_j^2$ if $j \leq i$. Vectors $\vec{r}^1$ and $\vec{r}^2$ are lexicographically equal if $\hat{r}^1 = \hat{r}^2$. Vector $\vec{r}^1$ is lexicographically less than $\vec{r}^2$, if $\vec{r}^2$ is lexicographically greater than $\vec{r}^1$. A utility allocation vector $\vec{r}$ is lexicographically optimal if it is feasible and if every feasible utility allocation vector is lexicographically less than or equal to $\vec{r}$. A lexicographically optimal utility allocation vector always exists, under our assumptions on the utility set. However, its computation is NP-hard. We proved this in [11], while considering layer allocation vectors in hierarchical networks with predetermined hierarchical structure and every layer consuming the same bandwidth irrespective of the source and the particular layer. This is a special case of what we consider here (we can look upon allocated layers as utility of a bandwidth). So the result holds in this case as well.

We suggest an alternate notion of fairness, maximal fairness. We use the concept of *relative fairness* introduced in [9]. A utility allocation vector $\vec{r}^1$ is *fairer* than another utility allocation vector $\vec{r}^2$, if for every virtual session i which has greater utility under $\vec{r}^2$ than under $\vec{r}^1$, there is some other virtual session j whose utility was already no more than that of i under $\vec{r}^1$, and has been decreased further by $\vec{r}^2$. A more formal definition of relative fairness follows.

A utility allocation vector $\vec{r}^1$ is *fairer* than another layer allocation vector $\vec{r}^2$ if

- $\vec{r}^1 \neq \vec{r}^2$ and
- if there exists an i such that $r_i^1 < r_i^2$, then there exists a j such that $r_j^1 \leq r_i^1$ and $r_j^2 < r_j^1$ (r_k^l is the kth component of $\vec{r}^l$, $l = 1, 2$.)

A utility allocation vector $\vec{r}$ is maximally fair if it is feasible and if no other feasible utility allocation vector is fairer than $\vec{r}$.

As we show later, maximally fair utility allocation exists and can be computed in polynomial complexity. But, first we investigate, whether maximal fairness is a good notion of fairness. Note that from the definition of maximal fairness, it is clear that if a utility allocation is maximally fair, then we can increase the utility of a receiver s, only by decreasing that of another receiver to a value below that of the new utility value of receiver s. Thus we can increase the utility of s only by being unfair to some other receiver. Also, if maxmin fair vector exists in any feasible set, then it is the only maximally fair vector[11]. So any algorithm for computation of a maximally fair utility allocation will yield a maxmin fair utility allocation, if one exists. Besides, any lexicographically optimal vector is maximally fair in any feasible set[11]. So lexicographically optimal vector is one of the maximally fair ones, in any feasible set. Thus maximal fairness is a good notion of fairness.

Maximum utility constraints can be incorporated by adding artificial links between receivers with maximum utility constraints and the rest of the network. Capacity of such an artificial link is equal to the bandwidth consumed by the maximum number of utilities of the respective receiver. So henceforth we shall ignore the maximum utility constraints.

4 Algorithm for Computation of Maximally fair Utility Allocation

Now we present an algorithm for computation of maximally fair utility allocation. First, we introduce certain terminologies. $\chi(s)$, $n(l)$ and $m(i,l)$, are defined in page 4 respectively.

Ψ_i is a real valued function defined as $\Psi_i(x) = \inf_{y \in A_i \cap (x,\infty)} y$, $\forall\, i$. Loosely speaking, $\Psi_i(x)$ is the smallest session i utility greater than x. If the utility set of session i, is a set of integers, then $\Psi_i(x) = \lceil x \rceil$ if x is not an integer and $\Psi_i(x) = x + 1$, otherwise. This function is well defined by property (1).

$\Phi_i(x)$ is a real valued function defined as $\Phi_i(x) = \max_{y \in A_i \cap [0,x]} y$. Thus, $\Phi_i(x)$ is the largest session i utility not exceeding x. If the utility set of session i, is a set of integers, then $\Phi_i(x) = \lfloor x \rfloor$. This function is well defined by property (2).

The minimum session link utility, μ_{il} is the maximum of the minimum utilities of session i virtual sessions traversing l, i.e., $\mu_{il} = \max_{s \in m(i,l)} \mu_s$.

L_s is the set of links traversed by virtual session s.

A virtual session s is saturated w.r.t. a utility vector $\vec{r}$ if it traverses a link l in which

$$\begin{aligned} r_s &= \lambda_{\chi(s)l} \\ \text{If } r_s = \Psi_{\chi(s)}(r_s),\ \sum_{i \in n(l)} \Upsilon_i\left(\lambda_{il}\right) &= C_l \\ \text{If } r_s < \Psi_{\chi(s)}(r_s),\ \sum_{i \in n(l)} \Upsilon_i\left(\lambda_{il}\right) &> C_l - \left(\Upsilon_{\chi(s)}\left(\Psi_{\chi(s)}(r_s)\right) - \Upsilon_{\chi(s)}\left(r_s\right)\right) \end{aligned}$$

A session is *saturated* on a link l if all the virtual sessions of the session traversing the link l are saturated.

$\eta_l(k)$ denotes the link control parameter of link l at the end of the kth iteration. Link control parameter is the iterate which would be used in computation of the maximally fair utilities. An utility equal to $\eta_l(k)$ would have been allocated to the unsaturated virtual sessions traversing the link, if there were no bandwith constraints on other links, and feasible utility allocations need only satisfy the capacity constraints.

$\eta_{il}(k)$ denotes the session link parameter of session i traversing link l. It is the utility assigned to session i, if there were no bandwidth constraints for any of its virtual sessions on other links and feasible utility allocations need only satisfy capacity and minimum utility constraints (i.e., utility set is the nonnegative real line).

$\omega_s(k)$ is the utility that is assigned to virtual session s at the end of the kth iteration if it is restricted to receive no more than any of its session link control parameters, $\eta_{\chi(s)l}(k)$, on its path. Here, it is the largest valid utility in $A_\chi(s)$ not exceeding, its minimum session link control parameter on its path.

$\Omega_{il}(k)$ is the utility allocated to session i in link l, under the utility vector $\vec{\omega}(k)$. $\Omega_{il}(k) = \max_{j \in m(i,l)} \omega_j(k)$.

$r_s(k)$ is the utility allocated to virtual session s at the end of the kth iteration. $\vec{r}(k)$ denotes the utility vector at the end of the kth iteration, with components, $r_s(k)$.

$\lambda_{il}(k)$ is the utility allocated to the session i in link l at the end of the kth iteration. It is actually the maximum of the utilities allocated to the virtual sessions in $m(i,l)$ at the end of the kth iteration.

$\Lambda(k)$ is the set of virtual sessions which are saturated w.r.t. utility allocation $\vec{\omega}(k)$.

$S(k)$ denotes the set of unsaturated virtual sessions at the end of the kth iteration.

$\Xi_l(k)$ denotes the set of unsaturated sessions traversing link l at the end of the kth iteration.

$F_l(k)$ denotes the total bandwidth consumed by the saturated sessions traversing link l at the end of the kth iteration.

The algorithm follows.

1. $k = 0$, $\eta_l(0) = 0$, $F_l(0) = 0$, $\Xi_l(0) = n(l)$ $\forall$ link l, $S(0) = \{1, \ldots, M\}$, $r_j(0) = \mu_j$, $\forall j \in S(0)$, $\lambda_{il}(0) = \max_{j \in m(i,l)} r_j(0)$.

2. $k \to k+1$

3. For every link l in the network compute the link control parameter. For this, first compute a variable $\alpha_l(k)$. If $\Xi_l(k-1) \neq \phi$, then $\alpha_l(k)$ is computed as follows:

$$\alpha_l(k) = \sup_\theta \{\theta : F_l(k-1) + \sum_{i \in \Xi_l(k-1)} \Upsilon_i\left(\max\left(\Phi_i(\theta), \lambda_{il}(k-1)\right)\right) \leq C_l\}$$

   (It can be shown that the above set is nonempty for all k.) If $F_l(k-1) + \sum_{i \in \Xi_l(k-1)} \Upsilon_i\left(\max\left(\Phi_i(\alpha_l(k)), \lambda_{il}(k-1)\right)\right) \leq C_l$, then link control parameter for a link l is $\eta_l(k) = \alpha_l(k)$, else link control parameter, $\eta_l(k)$ is chosen s.t. $\eta_l(k) < \alpha_l(k)$ and $\eta_l(k)$ satisfies some other technical properties. For all unsaturated sessions i passing through link l, $(i \in \Xi_l(k-1))$, $\eta_{il}(k) = \max(\eta_l(k), \lambda_{il}(k-1))$.

4. Compute $\omega_s(k)$ for all virtual sessions $s \in S$, where $\omega_s(k) = \Phi_{X(s)}(\min_{l \in L_s} \eta_{X(s)l}(k))$, if $s \in S(k-1)$, else $\omega_s(k) = r_s(k-1)$.
5. For every link l in the network, compute $\Omega_{il}(k)$ for every session i in $n(l)$, $\Omega_{il}(k) = \max_{s \in m(i,l)} \omega_s(k)$.
6. Compute the set of virtual sessions saturated during the kth iteration under utility vector $\vec{\omega}(k)$, $\Lambda(k)$.
7. If $\Lambda(k) \neq \phi$, compute the utilities allocated to the virtual sessions after the kth iteration, via, $r_s(k) = \omega_s(k)$, $\forall s$ and go to step (9).
8. Select a virtual session $s \in S(k-1)$ which satsfies *certain* technical properties and compute $r_j(k)$ for all virtual sessions j as

$$r_j(k) = \begin{cases} \omega_j(k) & j \neq s \\ \Psi_{X(j)}\left(\omega_j(k)\right) & \text{otherwise.} \end{cases}$$

9. For every link l in the network compute the session link utility in link l, for every session in $n(l)$, $\lambda_{il}(k) = \max_{s \in m(i,l)} r_s(k)$.
10. Compute the set of virtual sessions unsaturated after the kth iteration, $S(k)$.
11. If $S(k) = \phi$, i.e., all virtual sessions are saturated, the algorithm terminates, else go to the next step.
12. For every link l, compute the set of unsaturated sessions passing through link l at the end of the kth iteration, $\Xi_l(k)$
13. For every link l, for which $\Xi_l(k) \neq \phi$, compute the bandwidth consumed by the saturated sessions passing through link l, $F_l(k) = \sum_{i \in \Xi_l(k)} \Upsilon_i\left(\lambda_{il}(k)\right)$.
14. Go to step (2).

At every iteration k, the algorithm computes a "fair utility share" for every session i, the session link parameter, $\eta_{il}(k)$. Since a virtual session can receive utilities in utility set A_i only, utility $\Phi_i\left(\eta_{il}(k)\right)$ is offered to all virtual sessions of session i traversing the link. A virtual session is allocated the minimum utility offered on its path. If no virtual session saturates in the current iteration, then residual bandwidth may be used to increment the utility of some virtual session traversing this link and satisfying the properties mentioned in step (8). The utility of such a virtual session is increased to the next higher value. This is done because otherwise, the algorithm can continue forever. This is because in the next iteration, the same link control parameter will be computed and the process repeats again and again. A session is saturated if all its virtual sessions are saturated. The bandwidth consumed by the saturated sessions, if any, are computed. This bandwidth is subtracted from the link capacity, and the link control parameters are recomputed at the beginning of every iteration as per step (3) and the process continues. The algorithm determines which virtual session and sessions are saturated. A new iteration starts if still there are unsaturated virtual sessions. Upon termination, we have a maximally fair utility allocation.

We assume the bandwidth function Υ_i is right continuous and strictly increasing ($\Upsilon_i(x)$ is the minimum bandwidth required to attain session i utility x). Continuity is intuitive because if valid utilities in A_i are very close, then the minimum required bandwidths should also be very close. Strictly increasing property of the minimum bandwidth is also intuitive. This does not preclude the same utility corresponding to a range of bandwidth and hence non strictly increasing utility functions, because we only require the minimum bandwidth necessary for attaining a particular utility to be strictly increasing. The following results hold under this assumption.

Theorem 1 (Generalized Maximal Fairness) *If the generalized maximal fairness algorithm terminates, then the output utility allocation vector is*

1. *maximally fair,*
2. *maxmin fair, if a maxmin fair utility allocation exists.*

Theorem 2 (Finite-Termination Theorem) *The algorithm terminates in at most* $M + |\mathcal{L}|M^2$ *number of iterations, where* $\mathcal{L}$ *is the set of links and* M *is the number of virtual sessions.*

Every step of this algorithm has a complexity of $O(|\mathcal{L}|M)$. The algorithm must terminate in $M + |\mathcal{L}|M^2$ iterations. Thus the overall complexity of this algorithm is $O(|\mathcal{L}|^2M^3)$.

5 Conclusion and Discussion

Our algorithm is amenable to distributed implementation. The criteria for determination of utility of a virtual session uses information along the path of the virtual session mainly. The only place where the algorithm uses global information is that $r_s(k) = \Psi_{\chi(s)}\left(\omega_s(k)\right)$, for at most one virtual session, s. This feature of the algorithm is not crucial to the proof of maximal fairness of the output and is a matter of convenience. This increase in utility can be carried out for multiple virtual sessions, subject to feasibility and as long as they satisfy the criteria of step (8) and the algorithm will still output a maximally fair utility allocation.

In special cases, the algorithm terminates faster. If the utility functions are strictly increasing for all sessions, then $\Psi_i(x) = x$, for all i, x. In this case, the algorithm terminates in at most M iterations and yields a maxmin fair utility allocation. So the algorithm complexity is $O(\mathcal{L}M^2)$. If the utility function is a staircase function for every session i, and the ratio between the largest step jump for all sessions and the smallest step jump for all sessions is p, then the algorithm terminates in at most $M + p|\mathcal{L}|M^2$ iterations. So the algorithm complexity is $O(p\mathcal{L}^2M^2)$. We would like to point out that we made no assumptions on upper bound or lower bound for step jumps of utility functions.

References

[1] D. Bertsekas and R. Gallager, *Data Networks,* Englewood Cliffs, NJ:Prentice-Hall, 1987.

[2] T. Ballardie, P. Francis, and J. Crowcroft. Core based trees: an architecture for scalable interdomain multicast routing, *Proceedings of ACM SIGCOMM,* Sept. 1993, pp. 85-95

[3] Z. Cao and E. Zegura. Utility Max-Min: An Application-Oriented Bandwidth Allocation Scheme *IEEE INFOCOM'99*

[4] S. Deering and D. Cheriton. Multicast routing in datagram internetworks and extended LANs, ACM Transactions on Computer Systems, vol 8, no. 2, pp. 54-60, Aug. 1994.

[5] X. Li, S. Paul and M. Ammar. Layered Video Multicast with Retransmission (LVMR): Evaluation of Hierarchical rate control, *Proceedings of IEEE Infocom' 98*, March 1998

[6] X. Li, S. Paul and M. Ammar. Multi-session Rate Control for Layered Video Multicast *Technical Report* GT-CC-98 – 21, College of Computing, Georgia Institute of Technology, '98

[7] S. McCanne, Scalable Compression and Transmission of Internet Multicast Video. *PhD thesis*, Univ. of California, Berkeley, December 1996.

[8] S. McCanne, V. Jacobson and M. Vetterli. Receiver-Driven Layered Multicast, *Proceedings of ACM SIGCOMM '96*, Stanford, CA, September 1996.

[9] S. Sarkar and K. N. Sivarajan: Fairness in Cellular Mobile Networks *Proceedings of the 34th Annual Allerton Conference on Communication, Control and Computing*,1996, *pp* 457 – 469

[10] S. Sarkar and L. Tassiulas: Distributed Algorithms for Computation of Fair Rates in Multirate Multicast Trees *Technical Report* TR 99 – 42, Institute for Systems Research and University of Maryland, 1999.

[11] S. Sarkar and L. Tassiulas: Fair Allocation of Discrete Bandwidth Layers in Multicast Networks *Technical Report* TR 99 – 43, Institute for Systems Research and University of Maryland, 1999.

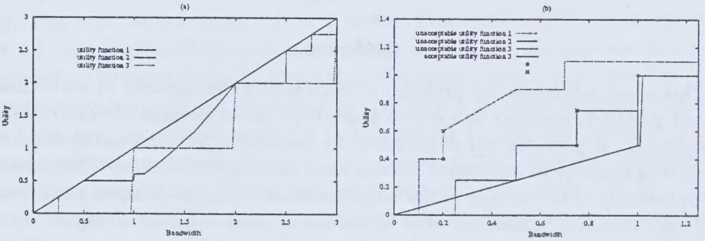


Figure 1: Figure (a) shows three utility functions. For first and third, utility of bandwidth x is the actual bandwidth which can be allocated when x units of bandwidth is available. Utility function 1 models the case when hierarchical structure is completely flexible and any bandwidth can be allocated, if it is available. The utility function is linear and the utility set is the nonnegative real line. Utility function 3 models the case when hierarchical structure is completely predetermined. Only a discrete set of rates can be allocated and the utility function is stair-case. Utility set is the set of nonnegative integers. Utility function 2 models the case when hierarchical structure is flexible in certain ranges of bandwidth and predetermined in other ranges. Here, the utility is the qualitative gain from the bandwidth. In the flexible region, function increases strictly, linearly or sublinearly. In the predetermined region, it is a stair-case function. Note that the step jumps are unequal, indicating unequal layer bandwidths for the same source. The utility set consists of closed intervals and some discrete points. In general, utility can also be the number of available layers when a ceratin bandwidth is available.
Figure (b) shows three unacceptable utility functions. The first utility function (unacceptable utility function 1) is not acceptable with data point 'o.' The utility set violates property (3). However the utility function becomes acceptable if the data point is replaced by the one marked, 'x'. Similarly, the second utility function (unacceptable utility function 2) is not acceptable with data point 'o', (property (4) is violated). Again the utility function becomes acceptable if the data point is replaced by the one marked, 'x'. The utility function for the third one (unacceptable utility function 3), with data point 'o' does not satisfy property (2). However, a small modification indicated by "acceptable utility function" satisfies property (2).

On the use of Multiple Working Points in Multichannel ALOHA with Deadlines*

Dror Baron[†]**and Yitzhak Birk**

Electrical Engineering Department

Technion — Israel Inst. of Technology

Haifa 32000, Israel

birk@ee, drorb@tx.technion.ac.il

Abstract

This paper addresses the problem of maximizing the capacity of multichannel Slotted ALOHA networks subject to a deadline and a permissible probability of exceeding it. A previous paper proposed to transmit a non-decreasing number of copies of a message in successive rounds until success or deadline. This yielded a low probability of failure due to the large maximum number of copies per message, with only minimal "pollution" due to the small mean number of copies. In this paper, we examine another way of implementing variable resource expenditure in different rounds: the channels are partitioned into groups, one for each round (until the deadline), and the channels used by later rounds are operated with lower offered loads. These Single-Copy Multiple-Working-Point (SC-MWP) policies are shown to attain a lower capacity than the optimal Multicopy Single-Working-Point (MC-SWP) scheme. Combining the two to form an MC-MWP scheme slightly improves capacity over MC-SWP. The SC-MWP approach can be more attractive when using a single transmitter per station because, unlike MC, transmission time is not prolonged. Therefore, as the trend to lower orbit satellites continues, leading to lower propagation delays, multiple-working-point policies should become of more interest.

Keywords: Multichannel ALOHA, satellite networks, deadline, multiple working points.

1 Introduction

ALOHA [1] is the simplest access scheme because it does not require channel sensing or collision detection, but performs worse than more elaborate schemes when those are practical. An important use of ALOHA at present is by satellite ground stations, because the long propagation delay precludes timely channel sensing. It is used as the primary access scheme for short messages, and in order to reserve channels for long ones [2].

---

*This work was supported in part by the Information Superhighway In Space consortium, administered by the office of the Chief Scientist of the Israeli Ministry of Industry and Trade.

[†] Presently with the ECE Dept., Univ. of Illinois at Urbana-Champaign, dbaron@ifp.uiuc.edu.

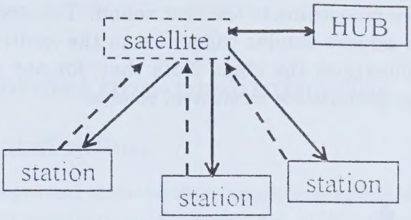


Figure 1: A typical hub-based satellite network.

Fig. 1 depicts a typical satellite-based ALOHA network. The *stations* transmit data in globally synchronized time slots over contention uplink channels (dashed lines). Successful reception by the *hub* is acknowledged by it immediately over contention-free downlinks (solid lines). The hub can be terrestrial or in space. If several simultaneous transmissions occur, they all fail. Stations can only learn about a collision through the absence of an acknowledgment. The time from the beginning of a transmission until the time by which an ACK for it must be received (or else it is considered to have collided) is referred to as a *round*. Unlike slots, which must be synchronized among the stations, a round is "private" and requires no coordination. A station retransmits packets until they succeed or until a deadline is exceeded. The typical duration of a round is up to 20 slots.

In a single-channel ALOHA network, retransmission delay must be randomized to prevent definite repeated collisions. To improve stability, a station must moreover increase the mean back-off time in later rounds. Current satellite networks employ as many as hundreds of channels. When operated with ALOHA, e.g., for small transactions, a station picks a channel at random for each transmission. The hub can receive concurrently over all channels. The randomized retransmission delay is replaced with immediate retransmission over a randomly chosen channel.

Over the years, the bulk of the research on ALOHA and related reservation schemes, e.g. [3], concerned maximizing capacity. Some attention was given to delay-throughput trade-offs and other performance measures. The advent of multichannel ALOHA networks has given rise to the use of redundant transmissions for performance improvement. For example, [4] studies *Multicopy ALOHA* (MC), whereby a station transmits several *copies* of a packet in each round, as a way of improving delay–throughput performance. We refer to the transmission of multiple copies per round as "redundancy" because, unlike retransmission upon failure, some of the transmissions may not be required.

Recently, Birk and Keren [5] proposed an optimization problem that reflects both intuitive user requirements and the desires of network designers: maximization of capacity subject to a deadline and a permissible probability of exceeding it. They proposed a *non-stationary Multicopy* (MC) transmission policy, whereby a station transmits a monotonically non-decreasing number of copies in successive rounds until successful reception or deadline. Dynamic programming was used to optimize the transmission sequence, resulting in a substantial increase in capacity relative to that attainable with classical ALOHA or even with (fixed) MC ALOHA [4]. The advantage is more pronounced for stricter constraints. They moreover adapted the optimized scheme to the practical situation wherein a station only has a single transmitter. This was done by transmitting a burst of copies in successive slots over randomly chosen channels, and then waiting for ACKs for all of them before proceeding to the next round. This technique, dubbed *Round Stretching*, was shown to achieve similar capacities to the multi-transmitter scheme in most situations. Fig. 2 illustrates the idea. Note that, for any given deadline, Round Stretching may reduce the permissible number of rounds.

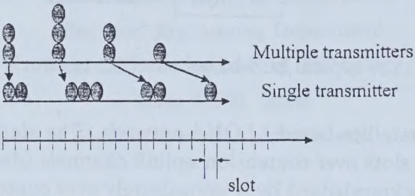


Figure 2: Round Stretching.

The non-stationary use of replication was extended in [5] to general erasure correcting codes for the case of single-round transmissions of multi-slot messages. In [6], such codes are considered for multi-round transmissions, as is the combination of such codes with reservations. The *Coding–Reservation* scheme of [6] raises capacity above $1/e$ by using a received message fragment to also request contention-free channels for the remaining fragment(s). The discussion in this paper is restricted to single-slot messages.

One can use *pure* MC policies, whereby the number of copies transmitted in any given round is deterministic (albeit not the same for all rounds), or *impure* policies whereby it is randomized. This idea is studied in [7] in the context of optimizing the throughput–delay trade-off with MC ALOHA. An impure variant of the replication-based scheme of [5] appears to only slightly increase capacity [6].

The key idea in the replication-based scheme of [5], which is also employed in this paper, is to permit a large maximum channel-resource expenditure per message while keeping the mean expenditure low. This is done by being more "wasteful" in the later rounds, which are less likely to take place. By so doing, the probability of failure can be made very low (because a message fails only after the maximum has been spent on it) without giving up much capacity. In [5], the expenditure manifested itself as speculative transmission of multiple copies in late rounds.

In this paper, we propose and study an alternative way of controlling the resource expenditure: the channels are partitioned into groups, one per round, with lower offered loads on the channels used for later rounds. (The number of channel groups equals the number of permissible transmission rounds until deadline.) This approach is dubbed *Multiple Working Points* (MWP). We begin by comparing the SC-MWP scheme (single copy per round) with the MC-SWP scheme of [5]. Then, the methods are combined into MC-MWP.

The remainder of the paper is organized as follows. In Section 2, we present the network model that is subsequently used for performance analysis, derive some preliminary mathematical relations for use in later sections, and define the policy design space. Section 3 proves that each round should best use a single working point and a deterministic number of copies. Section 4 provides performance analysis of MC-MWP policies, of which SC-MWP is a special case. Section 5 presents numerical results, and Section 6 offers concluding remarks.

2 Network model and preliminaries

2.1 Model and definitions

The network comprises ground stations that transmit single-slot messages over randomly chosen channels. A hub monitors all channels and ACKs all successful receptions. The lack of an ACK when it is expected indicates a collision. A station transmits in rounds, waiting for the results of one round before continuing to the next, until the deadline; then, an as-yet unreceived message is declared lost. (We will consider very small permissible loss probabilities, so "lost" messages may be reissued with a negligible effect.)

We assume an infinite number of stations and a large number of channels. The number of transmissions over any given contention channel in any given time slot is a Poisson random variable, independent from slot to slot and from channel to channel. Thus, the probability of collision of a packet is only a function of the offered load on the channel. This is an approximation, but the large number of channels along with the randomization make it close, and cause the degradation when used in finite networks to be graceful. Also, the inaccuracies are unlikely to distort the comparison among schemes.

ALOHA can be bistable in certain load regions. (It is never unstable because messages that miss the deadline are discarded by the sender.) However, the hub can detect such situations and "push" the network into the "good" stable point. The analysis in this paper applies to "good" stable operation.

A user-specified deadline is expressed in time units. For facility of exposition, we define this to be the time from the first transmission until the time of the latest transmission that would still arrive by the deadline. With fixed-size slots, D_s denotes the deadline in slots. For rounds of fixed duration, we use D_r to denote the maximum permissible number of rounds. P_e denotes the permissible probability of missing the deadline.

When Round Stretching [5] is used, let T_A denote the number of slots from single-slot transmission until ACK or until the next retransmission (round) may take place. Then,

$$D_s = (D_r - 1)T_A + N_{max}, \tag{1}$$

where N_{max} is the maximum total number of transmitted copies of a message transmitted. When $T_A \gg 1$, D_r is not affected much by N_{max}, and Round Stretching hardly changes performance. For small T_A, the effect varies.

Because messages may be dropped, albeit with a low probability, a distinction was made in [5] between the generation rate of messages, S_g, and the throughput S. Specifically, $S = (1 - P_e)S_g$. Derivation of channel capacity is complicated by the use of multiple channel groups with different working points. We therefore derive both S and S_g by examining the mean number of channel slots consumed by a message (successful or generated for S and S_g, respectively). Consequently, we can again write $S = (1 - P_e)S_g$. ("Channel slots" is a measure of channel resources, not delay.)

2.2 Useful relations

For pure MC-SWP policies [5],

$$G = S_g \cdot E[N], \tag{2}$$

where G denotes the per-channel offered load and $E[N]$ denotes the mean number of transmitted copies per message until success or deadline. Channel capacity is thus

$$S = S_g(1 - P_e) = \frac{G(1 - P_e)}{E[N]}. \tag{3}$$

The total number of copies transmitted per message is $N = \sum_i n_i \le \sum_{i=1}^{D_r} n_i \le N_{max}$, where n_i denotes the number of copies transmitted in round i. The probability of collision is $P_c = 1 - e^{-G}$. Since $P[\text{reach round } i] = (P_c)^{\sum_{j=1}^{i-1} n_j}$,

$$E[N] = n_1 + \sum_{i=2}^{D_r} n_i (P_c)^{\sum_{j=1}^{i-1} n_j}. \tag{4}$$

2.3 Design Space

The design space for single-slot messages has several dimensions: single/multiple copies per round; single/multiple working points; stationary/non-stationary; and pure/impure. A stationary policy acts identically in every round, whereas non-stationary ones are round-dependent. Pure policies are deterministic. Impurity of an MC-MWP policy can entail a probabilistic choice of the number of copies in any given round (with a possibly different mean for each round), as well as a probabilistic choice among several working points. Selection of a channel among those in the same group does not constitute impurity.

3 Optimality of a single working point per round

Numerical results have shown that, with multi-round MC-SWP policies, a probabilistic number of copies can increase capacity, but the increase is minute; also, a pure policy appears to be optimal for a single round [6]. In view of this, we only consider a deterministic number of copies in each round, denoted n_i, and prove that it is best to use a single working point for each round.

Theorem 1 *Any optimal policy that transmits a deterministic number of copies in each round uses a single working point for each round.*

Proof: Recalling the assumptions that were made, and that the fate of a transmitted copy is only influenced by the offered load (working point) of its channel, it suffices to consider a single round. The proof is by contradition. Consider two copies, each transmitted on a channel with a different working point; the probabilities of collision are denoted by P_{c_1} and P_{c_2}, respectively. We will show that transmitting them at a single WP, such that the probability that both collide is unchanged (equals $P_{c_1} \cdot P_{c_2}$), would reduce channel resource requirements.

Consider two working points with offered loads G_1 and G_2 copies per channel per slot. If the combined mean traffic over all channels operating at WP$_m$ is n_m copies per slot, the required number of channels is

$$W_m = \frac{n_m}{G_m}. \tag{5}$$

Therefore, the total number of channels required for the two working points in "support" of the transmission of a single copy at each of them is

$$W = \frac{1}{G_1} + \frac{1}{G_2}. \tag{6}$$

The same probability of error can be obtained by transmitting two copies at a single WP with collision probability

$$\tilde{P}_c = \sqrt{P_{c_1} P_{c_2}}. \tag{7}$$

According to (5), $\tilde{W} = 2/\tilde{G}$ channels are required. It suffices to show that $\tilde{W} \leq W$.

Consider $R(P_c)$, the revenue associated with a collision probability P_C. By defining $R(P_c) \equiv -\ln P_c$, an additive measure of revenue is derived. It is clear that $2 \cdot \tilde{R} = R_1 + R_2$. The channel requirement W can be calculated using the revenue as a parameter by

$$W(R) = -\frac{1}{\ln(1 - e^{-R})}. \tag{8}$$

It suffices to show that $W(R)$ is convex. It suffices to show that $G(R) = \frac{1}{W(R)}$, the offered load on channels with revenue R, is convex. But

$$\frac{\partial^2 G}{\partial R^2} = \frac{e^{-R}}{(e^{-R} - 1)^2} > 0. \tag{9}$$

Therefore, the channel requirement is convex in the revenue, so merging two copies at two different WPs lowers channel requirements. Consequently, among single-round policies that are pure on each of the WPs they use, only SWP policies can be optimal. □

4 Capacity of pure MC-MWP schemes

The probability of a message failing to meet the deadline is

$$P_e = \prod_{i=1}^{D_r} (P_{c_i})^{n_i}. \tag{10}$$

The generation rate at WP$_i$ is

$$S_{g_i} = \frac{G_i}{E[N_i]} = \frac{G_i}{n_i}. \tag{11}$$

The number of channels necessary for any given round is affected by the amount of messages entering it, which is lower if "cleaner" WPs are used in earlier rounds. In order to derive the network capacity (mean number of successful messages per channel slot), we must calculate the number of channels necessary for each WP, followed by the throughput obtained in each channel. Summing the throughputs and dividing by the total number of channels yields the mean per-channel capacity.

Consider W_1 channels used for WP$_1$ in the first round. The generation rate of distinct messages in the network is $S_{g_1} \cdot W_1$. The rate of messages entering round 2 is $S_{g_1} \cdot W_1 \cdot (P_{c_1})^{n_1}$. However, it is also equal to $S_{g_2} \cdot W_2$, so

$$W_2 = W_1 \frac{S_{g_1}}{S_{g_2}} (P_{c_1})^{n_1}. \tag{12}$$

Similarly, $S_{g_1} \cdot W_1 \cdot \prod_{k=1}^{i-1}(P_{c_k})^{n_k}$ messages enter round i, so

$$W_i = W_1 \frac{S_{g_1}}{S_{g_i}} \prod_{k=1}^{i-1}(P_{c_k})^{n_k}, i \geq 2. \tag{13}$$

The overall mean per-channel generation rate of distinct messages is

$$S_g = \frac{W_1 S_{g_1}}{W_1 + \sum_{i=2}^{D_r} W_1 \frac{S_{g_1}}{S_{g_i}} \prod_{k=1}^{i-1}(P_{c_k})^{n_k}}. \tag{14}$$

According to (11) and purity in round i, and since $P_{c_i} = 1 - e^{-G_i}$,

$$\begin{aligned} \frac{1}{S_g} &= \frac{1}{S_{g_1}} + \sum_{i=2}^{D_r} \frac{1}{S_{g_i}} \prod_{k=1}^{i-1}(P_{c_k})^{n_k} \\ &= \frac{n_1}{G_1} + \sum_{i=2}^{D_r} \frac{n_i}{G_i} \prod_{k=1}^{i-1}(1 - e^{-G_k})^{n_k}, \end{aligned} \tag{15}$$

and the capacity is $S = S_g(1 - P_e)$.

5 Numerical results

In order to compare the performance of MWP policies with SWP policies, a computer program that, given (n_i) and P_e, optimizes $\{G_i\}$ according to (15), was written. This was combined with an exhaustive search over (n_i) for the best values.

We begin by assessing the merits of SC-MWP as compared with tne conventional SC-SWP multichannel ALOHA. Next, we compare the two mechanisms for controlling resource expenditure, namely multiple working points (SC-MWP) and multiple copies (MC-SWP) [5] and offer some insights. Then, we discuss the MC-MWP hybrid. Up to this point, we assume an unlimited number of transmitters per station for the MC policies. Finally, results are presented for the practical single-transmitter situation, in which Round Stretching is employed whenever there are multiple copies per round.

5.1 Multiple transmitters per station

The results are summarized in Table 1.

Single-copy policies. Columns 3 and 6 depict the performance of SC-SWP and SC-MWP for several (P_e, D_r) constraints. The use of multiple working points is seen to yield a major performance boost.

Multiple copies vs. multiple working points. By columns 5 and 6, we see that MC-SWP [5] outperforms SC-MWP, so controlling the number of copies per round is superior to controlling the working points. This can be explained as follows.

With optimal MC-SWP policies [5], $P_e = (P_c)^{N_{max}}$, so

$$N_{max} = \frac{\ln P_e}{\ln P_c}. \tag{16}$$

Given a working point, N_{max} thus increases logarithmically with a decrease in P_e. Moreover, a message does not always utilize all the rounds, so $E[N]$ is even smaller and thus

Table 1: The capacity of MWP and SWP policies.

D_r	P_e	SWP			MWP		
		SC	Optimal MC		SC	Optimal MC	
		S	(n_i)	S	S	(n_i)	S
3	10^{-2}	0.190	1,2,4	0.279	0.233	1,2,4	0.281
	10^{-3}	0.095	2,3,7	0.247	0.158	1,2,6	0.248
	10^{-4}	0.045	2,3,10	0.233	0.110	2,3,9	0.234
5	10^{-2}	0.306	1,1,1,2,3	0.340	0.335	1,1,1,2,2	0.342
	10^{-3}	0.217	1,1,1,2,5	0.321	0.296	1,1,1,2,5	0.324
	10^{-4}	0.145	1,1,2,3,8	0.313	0.264	1,1,2,3,7	0.314

increases at most logarithmically with P_e. The cost, in terms of channels, needed to maintain a low error probability, is not very high.

With SC-MWP policies, the offered load on late-round channels is low, so

$$G \approx 1 - e^{-G} = P_c, \quad G \ll 1. \tag{17}$$

According to (5), the number of channels required for each "late" round is therefore roughly inversely proportional to the probability for channel collision in that round. The cost, in terms of channels, needed to maintain a low error probability, is quite significant.

MC-MWP policies. Permitting the use of multiple working points in conjunction with the optimal MC-SWP transmission sequence (n_i) of [5] slightly increases capacity. Joint optimization of (n_i) and the working points yields an additional small increase in capacity. Therefore, derivation of an optimal MC-MWP policy requires a search over the range of (n_i) in conjunction with the equations derived in the previous section.

Columns 7 and 8 of Table 1 show the optimal transmission sequence and the resulting capacity of MC-MWP. The improvements in capacity over that of MC-SWP (column 5) are below 1%. If capacity is the main design goal, MWP policies may not justify the higher implementation complexity.

5.2 MWP with Round Stretching

With Round Stretching, the permissible number of rounds D_r and the maximum number of copies N_{max} are related through (1). Consequently, especially for values of D_s that barely permit another round, one must decide whether to increase D_r at the cost of significantly reducing N_{max} or stay with one fewer round and slightly increase N_{max}.

Fig. 3 depicts channel capacity vs. D_s with Round Stretching for the policies considered in this paper. Results for MC-MWP with an unlimited number of transmitters and for Classical ALOHA are shown for reference. As D_s is increased and permits an additional round, the channel capacity with multiple transmitters rises immediately, whereas that with Round Stretching stays flat until such value of D_s for which an increase in D_r is warranted. Then, capacity rises sharply and approaches that with multiple transmitters per station. The MWP policies, due to their ability to use "clean" last rounds, cope well with constraints on N_{max}. Therefore, when optimized, they elect to use an additional round earlier (at smaller values of D_s) than MC-SWP. For this reason, MWP policies can save several slots of delay in Round Stretching. Note that MC-MWP and MC-SWP policies have identical performance up to $D_s = 7$, because a single round is used and the same (single) WP is thus chosen. Also, observe that capacity rises quickly with an increase in the number of rounds. Finally, though not shown in the figure, the performance with Round Stretching is closer to that with multiple transmitters for longer propagation delays (large T_A).

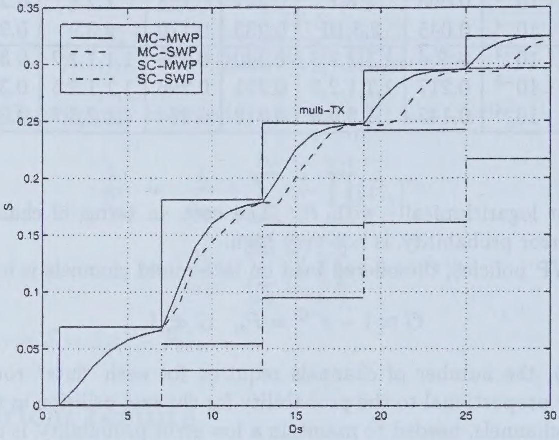


Figure 3: Channel capacity with Round Stretching. $P_e = 10^{-3}$; $T_A = 5$.

Since MC-MWP policies only provide slightly higher capacity than MC-SWP policies, we decided to check how many slots of delay can be saved using MWP policies, while attaining at least the same capacity as the optimal MC-SWP policy [5]. We held D_r constant, and checked by how much N_{max} can be decreased. Table 2 shows that MWP policies can provide a significant savings in N_{max}. When stricter delay constraints are used, N_{max} rises, as does the savings in slots.

Table 2: MWP slot savings vs. optimal MC SWP policies [5].

D_r	P_e	SWP			MWP			Savings
		(n_i)	N_{max}	S	(n_i)	N_{max}	S	
3	10^{-2}	1,2,4	7	0.279	1,2,3	6	0.280	1
	10^{-3}	2,3,7	12	0.247	1,2,5	8	0.247	4
	10^{-4}	2,3,10	15	0.233	2,3,7	12	0.233	3
5	10^{-2}	1,1,1,2,3	8	0.340	1,1,1,1,2	6	0.341	2
	10^{-3}	1,1,1,2,5	10	0.321	1,1,1,2,3	8	0.323	2
	10^{-4}	1,1,2,3,8	15	0.313	1,1,2,2,5	11	0.313	4

6 Conclusions

This paper focused on the maximization of capacity for single-slot messages in multichannel Slotted ALOHA networks. It explored the use of different working points in different rounds as a means of implementing a non-stationary expenditure of network resources in order to achieve low probabilities of failure while holding down the mean per-message resource expenditure. Through numerical results as well as some analytical insight, this Multiple-Working-Point approach was shown to be generally inferior to controlling the number of copies per round. (Nonetheless, it is significantly advantagous over the conventional SC-SWP approach.) An MC-MWP hybrid offers only a slight advantage when a station is equipped with multiple transmitters, but this advantage increases in the case of a single transmitter and Round Stretching, especially when the permissible delay and the permissible probability of failure are small. As the trend to lower orbit satellites and thus to lower propagation delays (fewer slots per round) continues, MWP policies should become of greater interest.

Directions for future research include the use of MWP policies for multislot messages, the combination of *Coding–Reservation* schemes [6] with MWP policies, and multiple service categories. Yet another direction is consideration of stability within the core access scheme.

Finally, we note that the results of this paper serve as yet another example of the benefits gained from the judicious use of redundancy for performance enhancement.

References

[1] N. Abramson,"Development of the ALOHANET", *IEEE IT*, vol. 31, no. 3, pp. 119-123, Mar. 1985.

[2] R. Rom and M. Sidi, *Multiple Access Protocols.* New York; Springer-Verlag, 1990.

[3] S. S. Lam,"Packet Broadcast Networks - A Performance Analysis of the R-ALOHA Protocol", *IEEE Trans. Computers*, vol. C-29, no. 7, pp. 596-603, July 1980.

[4] E. W. M. Wong and T. S. P. Yum,"The Optimal Multicopy Aloha", *IEEE Trans. Auto. Control*, vol. 39, no. 6, pp. 1233-1236, June 1994.

[5] Y. Birk and Y. Keren,"Judicious Use of Redundant Transmissions in Multi-Channel ALOHA Networks with Deadlines", *IEEE JSAC*, vol. 17, no. 2, pp. 257-269, Feb. 1999.

[6] D. Baron and Y. Birk, "Multiple working points and multi-slot message coding in multichannel ALOHA with deadlines", EE Tech report No. 1220 (also CC-pub 292, Technion, Haifa Israel, Sep. 1999.

[7] Y. W. Leung,"Generalised multicopy ALOHA", *Electronics Letters*, vol. 31, no. 2, pp. 82-83, Jan. 1995.

Specification and Analysis of a Reliable Broadcasting Protocol in Maude*

Grit Denker†, J.J. Garcia-Luna-Aceves‡, José Meseguer†,
Peter Csaba Ölveczky†, Jyoti Raju‡, Brad Smith‡,
Carolyn L. TalcottII
† Computer Science Laboratory, SRI International, Menlo Park, CA 94025, USA
{denker,meseguer,peter}@csl.sri.com
‡ Computer Engineering Department, University of California, Santa Cruz, CA 95064, USA
{jj,jyoti,brad}@cse.ucsc.edu
II Computer Science Department, Stanford University, Palo Alto, CA 94305, USA
clt@sail.stanford.edu

1 Introduction

The increasing importance, criticality, and complexity of communications software makes very desirable the application of formal methods to gain high assurance about its correctness. These needs are even greater in the context of active networks, because the difficulties involved in ensuring critical properties such as security and safety for dynamically adaptive software are substantially higher than for more static software approaches.

There are in fact many obstacles to the insertion of formal methods in this area, and yet there is a real need to find adequate ways to increase the quality and reliability of critical communication systems. As a consequence, in spite of the existence of good research contributions in formal approaches to areas such as distributed algorithms and cryptographic protocols, in practice new systems are developed for the most part in a traditional engineering way, using informal techniques, and without much to go by before detailed simulations or an actual implementation except for pseudocode and informal specifications.

The present work reports on an ongoing case study in which a new reliable broadcasting protocol (RBP) currently under development at the University of California at Santa Cruz (UCSC) has been formally specified and analyzed, leading to many corrections and improvements to the original design. Indeed, the process of formally specifying the protocol, and of symbolically executing and formally analyzing the resulting specification, has revealed many bugs and inconsistencies very early in the design process, before the protocol was implemented.

RBP performs reliable broadcasting of information in networks with dynamic topology. Reliable broadcasting is not trivial when the topology of the network can change due

---

*Supported by DARPA through Rome Laboratories Contract F30602-97-C-0312, by DARPA and NASA through Contract NAS2-98073, by Office of Naval Research Contract N00014-96-C-0114, and by National Science Foundation Grants CCR-9505960 and CCR-9633363.

to failure and mobility. The aim is to ensure that all nodes that satisfy certain connectedness criteria receive the information within finite time, and that the source is notified about it. The protocol should furthermore incur as low latency and as few messages as possible.

The amount of effort required in the formal specification and analysis process has been moderate, and it has been relatively easy for the researchers at UCSC to learn and use the specification formalism involved, so that the formal specification task has indeed been carried out in a joint way by researchers at SRI International, Stanford, and UCSC.

Reflecting upon the reasons for the success of this experiment, we can mention the following:

- *Early insertion of the formal method.* In this way, maximum benefit can be obtained, since the design can be corrected very early, before heavy implementation efforts have been spent.
- *Simplicity and intuitive appeal of the formalism.* The formalism involved, namely rewriting logic [12], is very simple and it is very well suited for specifying distributed systems, in which local concurrent transitions can be specified as rewrite rules.
- *Executability.* Rewriting logic specifications are executable in a rewriting logic language such as Maude [3]. This means that the formal model of the protocol becomes an *executable prototype*, that can be directly used for simulating, testing and debugging the specification.
- *Formal analysis.* Since the behavior of a communications protocol is highly concurrent and nondeterministic, a particular simulation run only exhibits *one* among many possible behaviors. Therefore, although direct execution can already reveal many errors and inconsistencies, a much greater confidence on the correctness of the design can be gained by formal analysis techniques in which *all* possible behaviors—up to termination, or up to a certain depth, or up to satisfaction of a specific state condition—are analyzed in detail. This can be done in Maude by means of exhaustive execution strategies that achieve a form of "model checking" analysis of the state space.

Formal Methodology

The formal methodology underlying our approach can be summarized by stating that *a small amount of formal methods can go a long way.* Approaches requiring full mathematical verification of a system can be too costly. Proof efforts should be used judiciously and selectively, carefully choosing those properties for which a very high level of assurance is needed. But there are many important benefits that can be gained from "lighter" uses of formal methods, without necessarily requiring a full-blown proof effort.

The general idea is to have a series of *increasingly stronger methods,* to which a system specification is subjected. Only after less costly and "lighter" methods have been used, leading to important improvements and corrections of the original specification, is it meaningful and worthwhile to invest effort on "heavier" and costlier methods. Our approach is based on the following, increasingly stronger methods:

1. *Formal specification.* This process results in a first *formal model* of the system, in which many ambiguities and hidden assumptions present in an informal specification are clarified.

2. *Execution of the specification.* If the formal specification is executable, it can be used directly for simulation and debugging purposes, leading to increasingly better versions of the specification.

3. *Formal model-checking analysis.* Errors in highly distributed and nondeterministic systems not revealed by a particular execution can be found by more a sophisticated model-checking analysis that considers all behaviors of a system from an initial state, up to some level or condition. In this way, the specification can be substantially hardened, and can even be formally verified if the system is finite-state.

4. *Narrowing analysis.* By using symbolic expressions with logical variables, we can carry out a symbolic model-checking analysis in which we analyze all behaviors not from a single initial state, but from the possibly infinite set of states described by a symbolic expression. Some of these analyses are already a special type of formal proof.

5. *Formal Proof.* For highly critical properties it is also possible to carry out a formal proof of correctness, which can be assisted by formal tools such as those in Maude's formal environment [4].

Up to now, we used methods 1–3 in the present case study, reaping important benefits from this use. In the future we may use methods 4 and 5 for selected purposes. The above case study is part of a broader effort to use rewriting logic and Maude in the context of active networks and network security. Related studies include: formal specification and analysis of cryptographic protocols [7]; work by Denker and Millen using Maude to specify the CAPSL and CIL specification languages [8]; and joint work with Carl Gunter and Yao Wang at the University of Pennsylvania on the formal analysis in Maude of an active-network algorithm written in the PLAN language [10].

The rest of the paper is as follows. In Section 2 we explain the key ideas of the RBP protocol and discuss its informal and formal specifications. In Section 3 we then discuss our use of methods 2–3 to further increase our assurance by means of execution and model-checking formal analysis. We finish the paper in Section 4 with some concluding remarks about our experience, and about future work.

2 Informal and Formal Specification of the RBP

2.1 Informal Specification

Little work exists on reliable broadcast protocols in dynamic networks. Most broadcast protocols are based on the PI and PIF protocols [17, 16], and all the broadcasting protocols for dynamic topologies extending these are based on the routing protocol by Merlin and Segall [11], which incurs too much communication to be attractive for, say, a wireless network [9]. It is the goal of our specification effort to come up with a more general approach (in terms of mobility of nodes) and more efficient protocols for reliable broadcasting in dynamic networks than exists so far. The starting point of our effort was an informal draft specification of the protocol given in [9]. We will refer to this description throughout the text as the original pseudocode.

In the following we illustrate part of the somewhat abbreviated version of the informal specification given in [9], which describes the proposed protocol for dynamic topologies.

The network is modeled as an undirected connected graph, where each node knows its neighbors but does not know anything more about the topology. In dynamic networks links can fail and come up. The objective of the protocol is to broadcast the message from a source node to all the nodes in the network and to give feedback to the source.

The dynamic case requires the handling of connecting and disconnecting network sections. If a new link (i, b) is established and i and b reside in different, unconnected parts of the network, then messages that have been sent to i must be forwarded to b and vice a versa. For example, if i has heard the latest message from source j, and b has not yet heard it because it had no physical path to j, then i forwards this message. There can be more than one message in a network. For example, different partitions of the network can have different messages which are in transit, and those messages might be forwarded to another partition when a new link comes up. Thus, each node has to store the latest number of a message with respect to a specific source node. Similarly, all other information, like the state of a node or its parent, is parametric in the source node. Thus, most of the node attributes have the source node as parameter. In particular, the following attributes form the data structure of a node: (1) a set of direct neighbors and several attributes which store information with respect to a given source node j. These are: (2) a flag which specifies whether the node is active or passive for j (i.e., a node is active if it has sent the latest message of j to its neighbors and has not yet received all acknowledgements); (3) a parent for messages from j, that is, the node from which it got first the latest message of source j; (4) the latest message number of messages from j; and (5) the state of a neighbor with respect to j, i.e., a flag which specifies whether the node is active or passive for a neighbor with respect to a message from j. A node i being active for a neighbor k with respect to a message from a source node j means that i has forwarded j's message to neighbor k but still waits for an acknowledgement from k.

Because of space limitations we only explain the behavior of network nodes upon the receipt of a message. The full specification tackles the cases of acknowledgements, link deletion and link addition as well, and can be found in [6].

The reaction of node i upon receipt of message m from p depends on the message number. In accord with the pseudocode suggested by in [9], we use the following notation to refer to attributes: N^i denotes the neighbors of node i (`nbs`), and for source node j ST^i_j denotes the state of node i (`state(j,_)`), s^i_j denotes the parent of i (`parent(j,_)`), SQ^i_j denotes the last sequence number of i (`seqNo(j,_)`), and ST^i_{jk} denotes the state of i with respect to neighbor k (`nbState(j,k,_)`). Figure 1 presents a pseudocode specification of the behavior later formalized in Maude for receiving a message: There are three main cases for receiving a message.

1. If a node repeatedly receives the current message (i.e., $SQ^i_j = m$), then it only replies with an acknowledgement to the sender.

2. If a node receives a newer message ($SQ^i_j < m$) or a message from a node for which it has no current message number (as it is the case if a new link comes up), then it stops an earlier diffusion source and starts a new one with the newer message. The node from which it got the latest message becomes the parent ($s^i_j := p$). It forwards the newer message to all neighbors other than p and waits for their acknowledgement. In case there are no more neighbors, it can acknowledge the receipt of the message.

3. If a node receives an older message ($SQ^i_j > m$), then its reaction depends on its state. If the node is still active, then it extends the ongoing diffusion computation

```
if $SQ^i_j = m$
  send ack$_m$ to $p$ for source $j$
if ($SQ^i_j < m$ or $SQ^i_j$ is not defined)
  $s^i_j := p$, $SQ^i_j := m$, $ST^i_{jp} := passive$
  if $i$ has at least one more neighbor other than $p$
    $ST^i_j := active$, $ST^i_{jk} := active$ for all $k \in N^i - p$,
    send message $m$ for source $j$ to all $k \in N^i - p$
  if $N^i = p$
    $ST^i_j := passive$, send ack$_m$ for source $j$ to $p$
if $ST^i_j = active$ and $SQ^i_j > m$
  $ST^i_{jp}$ := active, send message $SQ^i_j$ for source $j$ to $p$
if $ST^i_j = passive$ and $SQ^i_j > m$
  $ST^i_j := active$, $s^i_j := i$, $ST^i_{jp} := active$,
  send message $SQ^i_j$ for source $j$ to $p$.
```

Figure 1: Receiving a message.

by notifying the sender about the newest message. If the node is already passive, then it starts a local diffusion computation with itself as the source.

2.2 Formalization Process

There are many fuzzy requirements and much implicit knowledge in most informal specifications, such as in the starting point of our specification effort, a specification written in natural language and pseudo-code. The *formalization* part of the specification process aims at clarifying those requirements and making essential implicit knowledge explicit. For example, during the formalization process we realized that it was not clear whether one should take into account scenarios where an acknowledgment from node i to j may be received before the message sent from i to j. In other words, are messages/acknowledgments sent from i to j received in the same order as they are sent? This important aspect was not mentioned in the informal specification and provides an example of how the formalization effort helps to make implicit assumptions explicit. After discussing the matter it became clear that the implicit intention was in fact that the order of messages is preserved along a channel.

2.3 Maude Specification

The protocol is specified in *rewriting logic* [12, 13, 14, 15]. Rewriting logic is an executable logic which extends algebraic specification techniques to concurrent and reactive systems. Among its possible advantages over other executable specification formalisms are its being based upon a natural and well-known formalism, its natural integration of static and dynamic system aspects, the abstract modeling of communication, and its possibility to define execution strategies in the logic itself [1, 3, 2, 5]. All these advantages are fully supported by the Maude rewriting logic language [3]. Rewriting logic seems particularly suitable for specifying communication protocols, including those used for security. Such protocols are complex enough to warrant prototyping and their operational nature fits very well with the executable character of rewriting logic. The use of rewriting logic for the specification and analysis of security protocols is shown in [7].

Maude Basics

We briefly outline the syntactic key features of Maude that are essential for our case study. We model a distributed system configuration as a soup (multi-set) of concurrent objects and messages that behave according to a set of rewrite rules describing the behavior of individual objects. Maude allows to declare object-oriented *classes* with the following syntax

```
class C | a1: S1, ... , an: Sn
```

where `C` is the name of the class, and each `ai` is an *attribute identifier*, and `Si` is the sort inside which the values of the attribute identifier `ai` must range. An *object* in a given state is represented as a term `<O: C|a1:  v1, ...  , an:  vn>`, where `O` is the object's name or identifier, `C` is its class, the `ai`'s are the object's attribute identifiers, and the `vi`'s are the corresponding *values*. Message-constructing operators are introduced with the keyword `msg`. We can think of a system as a "soup" in which objects and messages float, so that any objects and messages can at any time come together and participate in a concurrent transition corresponding to a communication event of some kind.

Defining Network Nodes and Messages

We define the following class in Maude, which specifies the nodes of the network.

```
  class Node | nbs : OidSet, states : IdStatusPFun,
               parents : IdIdPFun, seqNos : IdIntPFun,
               nbsStates : IdIdStatusPFun .
```

The set of neighbors is a set of object identifiers. The other attributes are partial functions, that is, for a given source node identity, they deliver the state (`states : IdStatusPfun`), the identity of the parent (`parents : IdIdPFun`), the latest sequence number (`seqNos : IdIntPFun`), or, for each neighbor the state of the neighbor (`nbsStates : IdIdStatusPFun`). We modeled these data types in Maude by defining pairs and tuples which have as the first parameter always the identity of the source node. Partial functions are sets of pairs or triples. A variety of constants and operators is defined for these sorts. Details can be found in [6].

Assume the following node instance:

```
 < i : Node | nbs : set(k,l), states : state(j,active),
              parents : parent(j,k), seqNos : seqNo(j,5),
              nbsStates : nbState(j,l,active) >
```

Node i has two neighbors, k and l. Currently, i is active in a diffusion computation for source j. The latest message it got from j has the number 5, and i already forwarded this message to its neighbor l without having yet received an acknowledgement. Thus, i is in an active neighbor state for l with respect to source j.

The main messages in the system are: (1) sending and acknowledging a message (number) from a source node between network nodes; and (2) establishing and deleting links between nodes. We can formalize this in Maude by defining messages:

```
msg msg_To_From_Src_ : MachineInt Oid Oid Oid -> Msg .
msg ack_To_From_Src_ : MachineInt Oid Oid Oid -> Msg .
msg newlink : Oid Oid -> Msg .
msg failure : Oid Oid -> Msg .
```

In the following, to give a flavor for how RBP is formalized, we describe one of the rules for the dynamic RBP. The full specification is given in [6].

Receiving a message

As we pointed out before, there are three main cases for receiving a message.

If a node repeatedly receives the current message (i.e., $SQ^i_j = m$), then it only replies with an acknowledgement to the sender. This corresponds to the Maude rule

```
 rl [RepeatRecCurrentMsg] :
    < A : Node | nbs : nbs(B,OIDSET),
                 states : IDSTATUSPFUN,
                 parents : IDIDPFUN,
                 seqNos : seqNos(seqNo(C,M),IDINTPFUN),
                 nbsStates : IDIDSTATUSPFUN >
    (msg M To A From B Src C)
   =>
    < A : Node | nbs : nbs(B,OIDSET) >
    (ack M To B From A Src C) .
```

The lefthand side of the rule defines the precondition for the firing of the transition. In the above rule we require that at least one node and one message have to be in the network. The node with name `A` (`A` is a variable) has a neighbor `B` and the latest message number `A` heard from a source `C` was `M`. If `M` is also the currently received message number from source `C`, then it reacts by sending an acknowledgement back to `B`. The receiving node doesn't change its state.

3 Analysis Techniques using Maude

3.1 Validation with the Default Interpreter

A major advantage of rewriting logic specifications is that they can be *validated* immediately by executing test cases to provide quick feedback on the specification. This prototyping possibility comes for free. We used this feature every time the specification was modified, and often encountered errors quite easily on quite simple test examples (such as a network of three nodes). In this validation effort, we executed the test cases using Maude's default interpreter, which simulated some arbitrary run of the protocol for a given initial state of a network. The validation effort helped eliminate errors of syntax and of thought; furthermore, the built-in Maude facilities for tracing an execution were useful for discovering where the error occurred. This validation and correction cycle led to substantial improvements on, and a clear formalization of, the basic ideas of the starting informal protocol. In particular, it was agreed upon that the following follows from the informal specification:

1. A node should acknowledge all its "siblings" when it has received something (message or acknowledgment) from all its neighbors.
2. A node should acknowledge its "parent" node when it has received acknowledgments from all its neighbors except the parent.

As a first form of analysis, our final version of the protocol based on these ideas was validated using Maude's default interpreter. The extensive testing always returned the expected (and hoped-for) result.

3.2 Formal Analysis of the State Space

To substantially increase our confidence in the specification before any costly attempt at a formal proof of correctness, the specification can be subjected to close formal analysis using the meta-programming features of Maude to explore all states and behaviors that can nondeterministically reached from an initial state. Since the specification should be terminating, one could apply a strategy that explores all possible rewrite paths from some given initial state. In particular, we wrote a strategy for finding every non-rewritable state reachable from the initial state. For non-terminating systems, this setting can be modified to give e.g., every state which is reachable in less than 50 one-step rewrites from some initial state.

We experimented with different, increasingly complex, versions of the protocol. For example, executing the specification using an exhaustive strategy on a clique with three nodes did *not* produce the hoped-for result, namely a singleton set of irreducible states. Instead, the set of irreducible configurations reachable from the clique of three nodes included a term which indicated a deadlock before the expected end of one round of the protocol. A simple analysis of the output explained the error. If a node a is both "grandparent" and "sibling" to a node b, then, according to the ideas underlying the protocol, there is a deadlock as follows: Node c cannot acknowledge its parent a before it has received the "parent-acknowledgment" message from its child b. Node b cannot acknowledge its parent c before it has received an acknowledgment from its sibling a. And a cannot acknowledge c before it has received something (which in this case only can be an "parent-acknowledgment" message) from c!

Using Maude's meta-programming features, the user may himself define the rewrite strategies, thereby analyzing the specification in various ways. Although we only needed the quite straightforward exploratory analysis to invalidate our first version, this capability is very useful for analyzing various executions and extracting the relevant information from these automatically. Using the information from the exploratory analysis, the group came up with a new improved version of the protocol.

4 Concluding Remarks

The effort of formally specifying RBP for dynamic networks using the executable specification language Maude brought to light several weaknesses of the original pseudocode [9]. However, the analysis is not yet finished. For the moment, the following problems were identified and solved:

1. As described above we could eliminate a deadlock situation in the given protocol.

2. The pseudocode on which we based our first specification of the dynamic RBP was incomplete. In several places it was not clear what are the assumptions about node attributes.

3. A description of initial state was missing, and so was the termination condition for the protocol.

4. Moreover, the original pseudocode was incomplete with respect to how attributes are updated.

5. Some cases were left out in the original specification.

6. Other essential errors could be detected using the strategies. For example, given a test scenario with three nodes a, b, and c where b has the neighbors a and c, a is a source nodes which sends message number one and the link between a and b breaks down. Running the protocol with this initial configuration using an exhaustive search strategy delivers three different states of which one is a correct state, the second one reveals an undesired behavior and the third one showed an error in the original pseudocode which has been corrected in the current version.

Details can be found in [6]. The analysis is not finished yet. Our current implementation works fine for general test cases using a default strategy. But abnormal behavior appears when we run the protocol with an exhaustive search strategy which traverse all possible behaviors. This test cases are currently under investigation and will lead to changes in the protocol pseudocode. Moreover, further test cases in which link additions and deletions are combined with sending one or more messages are currently under investigation.

We have some further suggestions concerning other cases in the pseudocode which we have not implemented yet. We are currently discussing those suggestions.

References

[1] M. Clavel. *Reflection in general logics, rewriting logic, and Maude.* PhD thesis, University of Navarre, 1998.

[2] M. Clavel, F. Duran, S. Eker, P. Lincoln, N. Martí-Oliet, and J. Meseguer. Metalevel computation in Maude. In *Proc. 2nd Intl. Workshop on Rewriting Logic and its Applications*, Electronic Notes in Theoretical Computer Science. Elsevier, 1998.

[3] M. Clavel, F. Duran, S. Eker, P. Lincoln, N. Martí-Oliet, J. Meseguer, and J. Quesada. *Maude: Specification and Programming in Rewriting Logic.* Computer Science Laboratory, SRI International, Menlo Park, 1999. `http://maude.csl.sri.com`.

[4] M. Clavel, F. Durán, S. Eker, and J. Meseguer. Building equational proving tools by reflection in rewriting logic. In *Proc. of the CafeOBJ Symposium '98, Numazu, Japan.* CafeOBJ Project, April 1998. `http://maude.csl.sri.com`.

[5] M. Clavel and J. Meseguer. Reflection and strategies in rewriting logic. In J. Meseguer, editor, *Proc. 1st Intl. Workshop on Rewriting Logic and its Applications*, volume 4 of *Electronic Notes in Theoretical Computer Science.* Elsevier, 1996.

[6] G. Denker, J. J. García-Luna-Aceves, J. Meseguer, P. C. Ölveczky, Y. Raju, B. Smith, and C. Talcott. Specifying a reliable broadcasting protocol in Maude. Technical report, Computer Science Laboratory, SRI International, Menlo Park, 1999. Available at `http://www.csl.sri.com/~denker/pub_99.html`.

[7] G. Denker, J. Meseguer, and C. Talcott. Protocol Specification and Analysis in Maude. In N. Heintze and J. Wing, editors, *Proc. Workshop on Formal Methods and Security Protocols, 25 June 1998, Indianapolis, Indiana*, 1998.

[8] G. Denker and J. Millen. CAPSL Intermediate Language. In N. Heintze and E. Clarke, editors, *Workshop on Formal Methods and Security Protocols (FMSP99)*, Trento, Italy, 1999. Available at `http://www.csl.sri.com/~denker/pub_99.html`.

[9] J. J. Garcia-Luna. Reliable broadcasting in computer networks, 1998. Manuscript, UC Santa Cruz.

[10] M. Hicks, P. Kakkar, J. T. Moore, C. A. Gunter, and S. Nettles. PLAN: A Packet Language for Active Networks. In *Proceedings of the Third ACM SIGPLAN International Conference on Functional Programming Languages*, pages 86–93. ACM, 1998.

[11] P. M. Merlin and A. Segall. A failsafe distributed routing protocol. *IEEE Trans. Commun.*, 27(9):1280–1288, 1979.

[12] J. Meseguer. Conditional rewriting logic as a unified model of concurrency. *Theoretical Computer Science*, 96:73–155, 1992.

[13] J. Meseguer. A logical theory of concurrent objects and its realization in the Maude language. In G. Agha, P. Wegner, and A. Yonezawa, editors, *Research Directions in Concurrent Object-Oriented Programming*, pages 314–390. MIT Press, 1993.

[14] J. Meseguer. Rewriting logic as a semantic framework for concurrency: a progress report. In *Proc. Concur'96*, volume 1119 of *Lecture Notes in Computer Science*. Springer-Verlag, 1996.

[15] J. Meseguer. Research directions in rewriting logic. In U. Berger and H. Schwichtenberg, editors, *Computational Logic, NATO Advanced Study Institute, Marktoberdorf, Germany, July 29 - August 6, 1997*. Springer-Verlag, 1998.

[16] A. Segall. Distributed network protocols. *IEEE Trans. Info. Theory*, 29(1):25–35, 1983.

[17] A. Segall and B. Awerbuch. A reliable broadcast protocol. *IEEE Trans. Commun.*, 31(7):896–901, 1983.

Modeling and analysis of active messages in volatile networks[*]

C. Okino G. Cybenko
(cokino@dartmouth.edu) *(gvc@dartmouth.edu)*
Thayer School of Engineering
Dartmouth College
Hanover, NH 03755

Abstract

Interest in and development of mobile agent software systems has burgeoned in the past five years. Code mobility has many attractive attributes for performance and dynamic deployment of new distributed computing and information management applications. An *active message* is a datagram encapsulated as a mobile agent. The agent is persistent in the network, moving from node to node under its own internal routing logic and control at the application layer.

Active messages are particularly attractive in networks that have very unreliable links such as wireless networks in which the nodes are mobile. Such networks experience frequent link failures and other changes in topology. Active messages allow data to propagate between nodes that may never have viable TCP/IP type connections.

In spite of the growing implementation interest in mobile agents and active messaging, there are essentially no analytic models or results dealing with their performance. This paper presents a simple model for active messages in a network with frequent link failures. Using this model, we develop expressions for the expected delivery time of an active message along one path as well as expected delivery time for duplicated messages traversing disjoint paths between source and destination nodes.

1 Introduction

Mobile agents are programs that can migrate from computer to computer under their own control, carrying some part of their execution state with them. Mobile agent systems have been developed for a variety of languages and mobility models [6]. The original motivation for this programming paradigm was to make certain distributed information processing tasks more efficient by moving code *to* the data when the data is large relative to the expected "answer" obtained from processing the data. For example, a unique search of a large, remote database could be efficiently done by moving a relatively small

---

[*] This research was partially supported by National Science Foundation grant CCR-98113744, Air Force Office of Scientific Research Multidisciplinary University Research Initiative Grant F49620-97-1-0382,and DARPA Contract F30602-98-2-0107.

been investigated yet. This scenario presents challenges with respect to efficiently transporting, installing and executing the protocol code at the network nodes over the wireless links to deliver adequate performance for real-time traffic. We address the problem of transporting and installing the code at the network nodes through the use of an out-of-band transport mechanism. Real-time execution performance is delivered through the use of code objects that contain implementations of the protocols in platform-specific binary code.

The outline of the paper is as follows. The next section discusses related work in the area of active networking. The section on implementation describes details of the design and implementation of the active network, software switches and the congestion control programs used for demonstrating the effectiveness of the proposed architecture. In the evaluation section, we discuss performance measurements carried out to evaluate the network and the results obtained. The paper ends with some conclusions and a list of references used.

2. Background and Motivation

Carrying mixed real time (audio) and non-real time (data) traffic over low speed, wireless links using standard ATM protocol presents many problems. The two principal problems are the high error rates due to channel fading, and the transmission latency of the cells. The modifications made to improve performance resulted in a Narrowband ATM [2] or NATM protocol. NATM uses a cell length of 24 payload bytes and 2 header bytes, as opposed to the standard 53-byte cell. Additionally, it utilizes a geometric Hamming Code based Forward Error Correction scheme that adds 50% redundancy to each cell. The smaller cell size reduces the transmission latency to acceptable levels for real-time traffic, and the FEC helps reduce errors to acceptable levels without retransmissions bringing down throughput.

Real-time traffic like audio is delay-sensitive as well as jitter-sensitive. The quality of service for such traffic is usually obtained in ATM networks by treating it as Constant Bit Rate (CBR) traffic and reserving resources for it. But the quality comes at a price. Assume the case of a low bandwidth connection being used for audio as well as data communication. Assigning a significant chunk of the already scarce bandwidth of the wireless link to the audio traffic to ensure zero drop-off in sound quality results in lower throughput for the data connection, which is typically bursty and lasts for a short interval. Audio data lends itself to compression and transcoding with/without source interaction. This suggests that audio traffic can be sent at a lower rate (lower bits per sample) by some method of bandwidth reduction for the duration of the data transfer. If the remaining bandwidth can be assigned to the data connection, then the cumulative performance of the wireless link is likely to be higher for the integrated traffic.

Examples of active processing can be found in existing congestion control mechanisms. Recognizing the performance degradation caused by fragmentation [3], packet level discard techniques have been explored to improve the performance of TCP/IP over ATM [4] [5]. The Partial Packet Discard strategy drops all cells in a packet that follow a dropped cell [4]; the Early Packet Discard strategy improves performance by aggressively dropping entire packets when a congestion threshold is reached [5]. The presence of a cell loss priority bit in the ATM cell header allows the source to indicate to the network that some cells should be treated differently than others under congestion.

Active networking has been applied to tackle the problems of congestion. Researchers at Georgia Institute of Technology [6] have targeted congestion control in ATM networks using active networking. They follow a functional paradigm in which packets select from a pre-defined set of active networking functions that have been pre-programmed into ATM switches in the network. While this paradigm provides more control over the packet's execution, it restricts dynamic customization of the network because it is impossible for the network to anticipate all possible requirements of user applications.

In a Turing machine model of an active network, nodes are programmable elements that allow applications to execute user-defined programs at the nodes. Therefore, nodes in an active network are called **active nodes**. Active networks exchange ***SmartPackets***, which contain a destination address, user data, and methods that are executed locally at any node in the active network. Active nodes perform the functions of receiving, scheduling, executing, monitoring and forwarding SmartPackets. New services and protocols are injected into the network by sending the protocol code within SmartPackets. Active nodes install the protocols by executing the SmartPackets.

An implementation of a selective filtering policy for wireless networks using the above approach has been discussed in [7]. In that effort, a terrain visualization application running over a wireless link panned the terrain by attempting to download image packets from a remote server. If congestion occurred, a filter installed at the wireless gateway selectively dropped packets that were outside the application's current view. However, transcoding delay-sensitive traffic such as audio in real-time and over wireless links presents many challenges. In an ATM network, the switches can execute an audio transcoding algorithm when they detect data transmission to reduce the bandwidth required by the audio traffic and provide better throughput to the data connection. To combat the bandwidth limitations of the wireless links, we use an out-of-band technique to install the requisite algorithms at the switches. This approach differs from the functional model used in [6]. Secondly, platform-specific binary code is used to implement the transcoding algorithm. This is unlike the approach used in [7], where the code object is in Java [8] bytecode format and the code is run in an interpreted environment. We use the binary code approach to achieve real-time performance. To implement this scheme in an active networking environment, we use the prototype active network toolkit called Magician [9], developed at the University of Kansas. In the next section, we describe an active congestion control scheme to handle real-time audio transcoding and the modifications made to the Magician toolkit to adapt it to deliver real-time performance over a Narrowband ATM network.

3. Active Congestion Control

ATM switches are usually built using custom hardware, since they are highly parallel and speed intensive applications. But in order to implement active functionality on an ATM switch, we choose to implement the switching capabilities in software so that the active processing can be integrated with the switching process. This provides a cost-effective solution for Narrowband ATM networks, without degradation in performance while utilizing the unused CPU cycles leftover during active processing by the active node manager The implementation of the software switch and its integration with the active node and the ATM subsystem is shown in Figure 1. The Q-port Signaler shown in the figure provides signaling stacks for a configurable number of ATM interfaces. The active manager is the Magician platform on the switch for providing active networking execution environment. ATM cells are switched by the MicroSwitch driver when no active processing is required i.e. no congestion control is required. However, when active processing is required, ATM cells are passed to the user space to be processed by the active manager before being switched out.

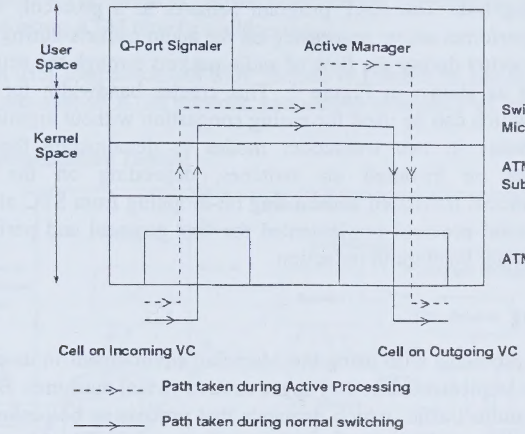

User Space
Kernel Space
Q-Port Signaler
Active Manager
Switching Subsystem: MicroSwitch Driver
ATM Driver Layer Subsystem
ATM Interface Cards
Cell on Incoming VC
Cell on Outgoing VC
Path taken during Active Processing
Path taken during normal switching

Figure 1. Block diagram of Software ATM Switch

To provide real-time performance on the low bandwidth links, we require a low bit-rate speech coder to encode the audio stream. Also, it is required that the speech coder support source-independent rate translation capability wherein it is possible to transform the bit rate of the coded speech stream at any intermediate node in the network, without interaction with the source. This feature is used in our congestion control scheme wherein the active nodes inside the network transcode the audio on congested links to lower bit rates to ease congestion. Embedded coders provide source-independent rate translation capability, where the parameters of the lower rate coder are embedded in the parameters of the higher rate coder [10]. To reduce the bit rate, the higher rate parameters are stripped off. The Sinusoidal Transfer Coder (STC) [11] allows inter-operability at different rates and source-independent rate translation using bit-dropping technique. STC is a multi-rate coder that operates at several discrete rates from 8 kbps to 2.4 kbps. In the current implementation, STC normally operates at 4.8 kbps. During congestion, STC performs bit-dropping and produces voice coded at 2.4 kbps. The data stream is not encoded since it is not be subject to active processing. Under congestion, data cells are dropped in the switch and transport mechanisms ensure reliability of data transfer.

Audio is encoded and encapsulated using Robust Audio Tool (RAT). The RAT is a network audio tool that allows users to participate in audio conferences over the Internet. Audio connections can be directly between two participants, or between a group of participants on a common multicast group. The RAT, as per IETF standards, uses RTP [12] over UDP/IP as its transport protocol, and conforms to the RTP profile for audio and video conferencing with minimal control. We implement a portion of the Internet draft [13], aimed towards compressing IP/UDP/RTP headers. In an ATM network, only RTP headers have to be compressed. Compression is done on a link-by-link basis as it provides better performance compared to an end-to-end basis, in terms of delay and loss rate. The design provides simplicity at the expense of generality in order to reduce the compression-decompression time.

When the RAT is operated in the transcoder mode it allows bandwidth translation between the incoming and outgoing link. The RAT program behaves as a protocol, which when installed on the switches performs active processing on the audio packets during congestion. Audio coding translation occurs during the flow of audio packets through the active manager through active processing as shown in Figure 1. This creates bandwidth for use on the outgoing congested link, which can be used for easing congestion without significant loss in audio quality. RAT operates in two transcoder modes to demonstrate that arbitrarily customized protocols can be installed on switches, depending on the connection requirements. The first protocol translated audio using bit-dropping from STC at 4.8 kbps to STC at 2.4 kbps. The second protocol implemented the first protocol and performed RTP header-compression for further bandwidth reduction.

3.1 Active Networking

One of the fundamental problems with using the Magician environment in its present form is that the active nodes are implemented in user space as Java virtual machines. But the target traffic contains real-time audio traffic, which demands that processing be performed at line rates to minimize delay and jitter. We had to modify certain features of the Magician code to integrate it with the software ATM switch. Here we discuss some of the modifications.

There are two major approaches towards transporting code in an active network, the out-of-band and the in-band approach. Magician uses the in-band approach in which the SmartPackets carry the protocol code as well as the data as they travel through the network. Network nodes process the packet by executing the accompanying code. In the out-of-band or discrete approach, the active nodes load and install the application-specific protocols during connection setup time. Data packets passing through these switches request processing by invoking one of the previously installed protocols. The Narrowband ATM Network uses low-bandwidth links to interconnect hosts. The size of the program code is significant enough to cause a long delay in transfer of the SmartPackets carrying audio data, if the in-band approach is used. Instead, we use an out-of-band loading scheme because that eliminates the latency of transporting and installing the code. The congestion control schemes are downloaded on to ATM switches in the connection path at connection set-up time.

The role of SmartPackets is different in this implementation. SmartPackets are used *only* to load program code in the active node. Since the out-of-band approach is used to load the program code using SmartPackets, these SmartPackets are sent out before the actual audio traffic is sent out. Each node on receiving the program code stores it locally in a cache from which it is retrieved when necessary. The program code will never be re-transmitted between same pair of interfaces, since each active node keeps track of the adjacent nodes that has received the particular program code previously.

The role of program code is different here as well. Program code, by itself, does not implement new protocols to be used by user-data. Instead, program code when executed causes execution of system-dependent binary code outside the Java virtual machine and the relevant parameters are passed to the binary code. The binary code in this case is used to implement congestion control algorithm. Thus, once SmartPackets install program code in the active node and the program is executed, the binary code is invoked. Other modifications made were:

- Magician identifies active nodes using IP host address and UDP port combination. In the Narrowband ATM Network, active nodes reside on software switches and are therefore identified using ATM interface addresses.
- Magician uses connectionless UDP sockets to transfer to and from the network. To make connections over an ATM network, connection-oriented PVC or SVC sockets are used.

3.2 Implementation Details

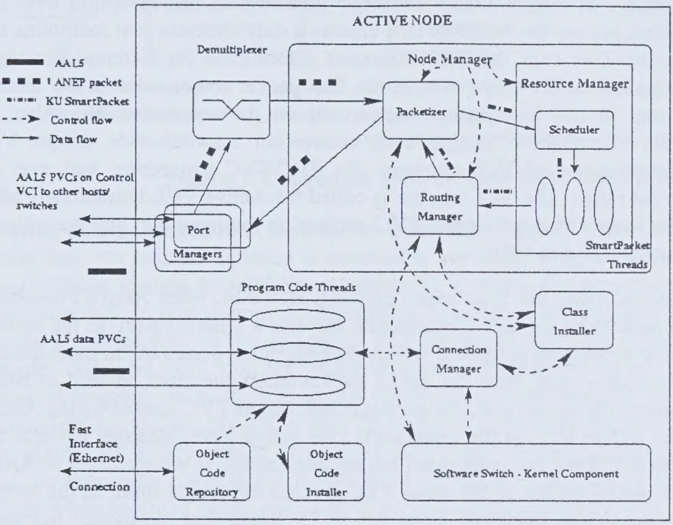

ACTIVE NODE
AAL5
ANEP packet
KU SmartPacket
Control flow
Data flow
Demultiplexer
Node Manager
Resource Manager
Packetizer
Scheduler
Routing Manager
SmartPacket Threads
AAL5 PVCs on Control VCI to other hosts/ switches
Port Managers
Program Code Threads
AAL5 data PVCs
Class Installer
Connection Manager
Fast Interface (Ethernet) Connection
Object Code Repository
Object Code Installer
Software Switch - Kernel Component

Figure 2. Active node architecture

The model of an active node in the Narrowband ATM network is shown in Figure 2. An active node is made up of (possibly) multiple processing environments such as a PLAN [14] execution environment or an ANTS [15] environment. For simplicity of demonstration, we assume that an active node contains only one execution environment. When an active node boots up, it starts a set of Port Managers, a Demultiplexer, and a Node Manager. There is one Port Manager per ATM interface on the end-host/switch. Usually, end-hosts have only one ATM interface, and hence only one Port Manager is started. A switch has one Port Manager for each enabled ATM interface.

The list of interfaces is obtained by querying the MicroSwitch through *ioctl()* calls using Java Native Interface (JNI). Since no information about the switch is available through standard system calls and Java does not provide support through its libraries, a library had to be created. Native functions were implemented which query the switch for different bits of information using *ioctl()* calls and return information to the Active Manager using Java Native Interface [16]. The Port Manager opens a PVC socket on a unique VCI (62) called Control VCI on each of the enabled interfaces. Since Java does not provide socket API for ATM, an API was defined and implemented which allowed AAL5 PVC socket management. The Port Managers read and write to these sockets and thereby communicate with immediate switch/host that is connected to the interface they are working with. The Port Manager on the corresponding host/switch listen on a socket opened on the same VCI. Incoming SmartPackets are queued at the Demultiplexer that verifies that the packet conforms to ANEP standard and then demultiplexes it to the correct processing environment. The byte stream constituting the packet is then handed over to the environment's Node Manager.

In the current implementation, the Node Manager starts other managers such as the Resource Manager, the Port Interface Manager, and the Connection Manager. In this implementation, there is no Routing Manager. The Port Interface Manager controls the assembly and disassembly of SmartPackets and their transmission and reception over the wire. During connection set-up, the MicroSwitch creates a data structure that maintains the state for the connection. This data structure maintains information for different functions such as whether the packet on the connection is the first packet or not, what is the special identifier that was sent in the first packet transmitted on the connection etc. Also, at connection set-up, the MicroSwitch assigns each connection a switch-wide unique VCI number from a reserved range of VCI numbers. No PVC/SVC connection will ever be assigned VCIs from that range. The VCI number is called the Active VCI. Immediately after connection set-up, the Active Manager opens PVC sockets on the input port and output port of the connection using the Active VCI.

Data is usually switched using the class-based queuing. However, when Active Processing is desired the AAL5 packets on the active connection traverse a different path. In the current implementation, Active Processing is desired whenever congestion occurs in the input queues for an output port. In such a case, after the AAL5 packet enters the input, instead of being queued in the input queue for output port, the packet is sent on the PVC opened by the Active Node Manager on the Active VCI on the input port. The Active Node Manager extracts the RTP packets from the AAL5 packets, processes the encoded audio, re-encapsulates in AAL5 format and writes the AAL5 packet to the same PVC it reads the packet from. In the switch kernel, the MicroSwitch driver interprets from the VCI number that the packet has been processed by the Active Node Manager and then queues up the packet on the input queue for the destined output port it is to be switched to. The path that the AAL5 packet then follows is the same that it would follow if there were no Active Processing. Figure 3 shows the diagrammatic implementation of the switching scheme.

The Connection Manager is responsible for checking creation of connections to the MicroSwitch and invoking binary code programs that implement congestion control processing on the user-data read through AAL5 PVCs. The Active Node Manager starts the Connection Manager. The Connection Manager polls the switch using native functions to determine if an active connection is set-up. Since active connections follow RTP standard, they always occur in pairs, one connection for the RTP data part and the other for the RTCP control part. Each active connection immediately after connection set-up stores a special identifier in every MicroSwitch in its connection path. This identifier in addition to the magic string "AN" contains the name of the class whose program code it wishes to execute on the connection. In this way the connection has control over what algorithm it wishes to implement when congestion occurs. Whenever an active connection is set-up, as mentioned an active VCI is also assigned to the connection. This is the VCI on which AAL5 PVC socket is opened by the binary code and is the VCI on which on which data is sent for active processing in the MicroSwitch before being switched out.

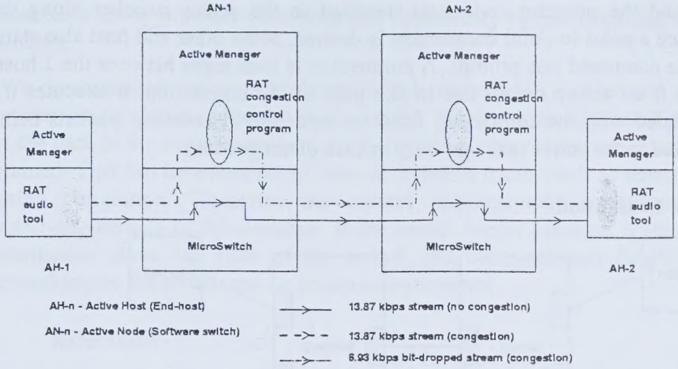

AN-1
AN-2
Active Manager
RAT congestion control program
Active Manager
RAT congestion control program
Active Manager
RAT audio tool
Active Manager
RAT audio tool
MicroSwitch
MicroSwitch
AH-1
AH-2
AH-n - Active Host (End-host)
AN-n - Active Node (Software switch)
13.87 kbps stream (no congestion)
13.87 kbps stream (congestion)
6.93 kbps bit-dropped stream (congestion)

Figure 3. Congestion control using bandwidth reduction

After an active connection is setup, the Connection Manager derives the program code to be executed. An internal database is checked to see if the program code has been installed by out-of-band loading through SmartPackets. If it exists, the program code is executed. The program code in turn attempts to execute the binary code it is required to invoke. If the binary code is not resident on the switch, an FTP request is made via a fast interface to fetch the binary code and store it on the switch. Program monitors have been implemented to ensure that all program codes wanting to execute the binary code wait for the single FTP request to complete and do not fire off more FTP requests. Once it is ensured that the binary code is available, the binary code is invoked by the program code by passing some arguments. One of the arguments contains the Active VCIs for the RTP and RTCP connections. The binary code when executed opens AAL5 PVC sockets on the Active VCI on the relevant interfaces. Thus, whenever an AAL5 packet arrives on the input side of an active connection, the MicroSwitch driver pushes it into the receive queue of the Active VCI socket for the active connection. The binary code reads the user-data from the AAL5 packet, processes it, re-encapsulates the data and sends it out on the socket. When the switch kernel receives data on an Active VCI socket, it is diverted to the MicroSwitch for switching out. Whenever an Active Connection is torn down, it is detected in the next Connection Manager poll and the Connection Manager stops the execution of the program code and binary code assigned to that connection.

3.3 Operation

During network setup, the active managers are initiated on all the switches. Depending on whether the machine is an end-host or a switch it starts the required managers e.g. end-hosts do not start the Connection Manager due to its obvious functionality. It is assumed that before active managers start up on end-hosts, all the user-specific protocols implemented using program code have been loaded appropriately. After the active manager starts up on the end-hosts, the user is prompted with a command-line prompt. The user keys in a command that includes the name of the class of the program code. This results in two things. First, the RAT application in started on that end-host. Second, the active manager sends the program code out on the PVC socket opened on a unique control VCI (62). The switch at the other end of the link receives the program code and stores it in its repository. It then sends the program code on the control VCI PVCs of all interfaces that are enabled on the switch. This results in a cascading effect and the program code gets installed in the active switches along the connection path. Since a point-to-point connection is desired, some other end-host also starts up the RAT using the command line prompt. A connection is then made between the 2 hosts desiring connection. If an active switch lies in the path of the connection, it executes the program code associated with the connection from its repository. However, packets being switched will be routed to the active manager only in case of congestion.

4. Performance Evaluation

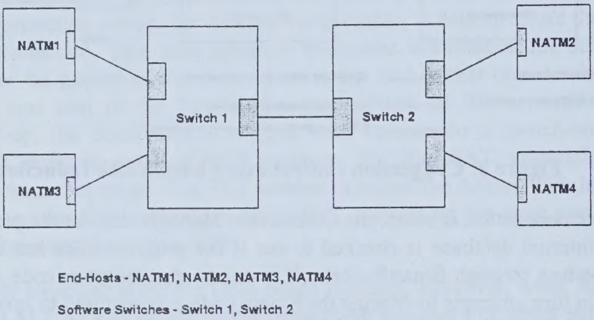

NATM1
NATM2
NATM3
NATM4
Switch 1
Switch 2
End-Hosts - NATM1, NATM2, NATM3, NATM4
Software Switches - Switch 1, Switch 2

Figure 4. Network topology

The efficacy of the scheme discussed above was evaluated using STC-coded audio. The network topology shown in Figure 4 was used to test the active networking functionality. The audio source initially produces 4.8 kbps coded audio. Two congestion control algorithms were tested. The first one was RAT in bit-dropping mode wherein the bandwidth of STC encoded audio was reduced from 4.8 kbps to 2.4 kbps. The second one was RAT in bit-dropping mode along with header compression to reduce the bandwidth on the congested link. Standard audio clips were sent from the transmitting end-hosts using RAT. Congestion was simulated on the switches causing active processing to be performed on the audio traffic. Audio clips were recorded on the receiving end-hosts. The performance of the clips using the different congestion algorithms was compared subjectively. It was observed that there was no perceivable drop-off in audio signal quality when congestion occurred and the audio traffic bandwidth was switched from 4.8 kbps to 2.4 kbps using the bit-dropping algorithm. When congestion was alleviated, the bit-dropping algorithm was not invoked and the audio bandwidth switched back to 4.8 kbps.

5. Conclusions

In this implementation it has been shown that an architecture which allows user-data to be processed within the switches in ATM networks, increases flexibility in protocol implementation within the network. An application packet has the freedom to select computation that occurs on its data as it flows through the network. This is especially useful in the current networks where integrated traffic is commonplace. Active networks enable customized protocols to be easily implemented. The use of the Java language to implement the active network makes it easy to deploy it on any computing system. The connection-oriented nature of ATM traffic allows the architecture to be scalable, as each interface needs to know only about the interface connected to the other end of the link. The time required for connection set-up provides an appropriate opportunity for pre-installing user protocols on the switches.

We demonstrated the applicability of active networks in narrowband networks by using it for reactive congestion control of audio traffic. Inspite of the additional processing performed on the data in the active managers on the switches, the real-time nature of audio traffic was retained. This can be extended to other multimedia traffic such as compressing digital video during congestion. However, the current implementation requires support from the underlying kernels in the switches. If the switch kernel provides a standard API that gives information about the state of the switch, the computational flexibility of using active networking on the switch can be immensely increased.

6. References

1. Tennenhouse, D. and Wetherall, D., "Towards an active network architecture," *Multimedia Computing and Networking*, January 1996.
2. Lindsley, E, *Narrowband ATM Networks*, M.S. Thesis, University of Kansas, 1997.
3. Kent C. and Mogul, J., "Fragmentation considered harmful," *ACM SIGCOMM*, 1988.
4. Armitage, G. and Adams, K., "Packet reassembly during cell loss," *IEEE Network Magazine*, September 1993.
5. Romanov, A. and Floyd, S., "Dynamics of TCP Traffic over ATM Networks," *IEEE Journal on Selected Areas of Communications*, May 1995.
6. Bhattacharjee, S., Calvert, K., and Zegura, E., "On active networking and congestion," *Technical Report GIT-CC-96/02*, Georgia Institute of Technology, 1996.
7. Kulkarni, A., and Minden, G., "Active networking services for wired/wireless networks," *IEEE INFOCOM '99*, 1999.
8. Gosling, J., and McGilton, H., *The Java Language Environment: A White Paper*, Sun Microsystems, 1995.
9. Kulkarni, A., Minden, G., Hill, R., Wijata, Y., Wahhab, F., Sheth, S., Gopinath, A., Nagarjan, A., and Pindi, H., "Implementation of an active network prototype," *IEEE OPENARCH '98*, 1998.
10. Champion, T., and Evans, J., "A flexible multirate speech coder," *Proc. Intl. Conf. On Signal Processing*, pp. 1440-1443, September 1993.
11. McAulay, R., and Quatieri, T., "Multirate sinusiodal transform coding at rate from 2.4 kbps to 8 kbps," *IEEE Intl. Conf. Acoustics, Speech, Signal Processing*, pp. 1645-1648, 1986.
12. Schulzrinne, H., Casner, S., Frederick, R., and Jacobson, V., "RTP: A transport protocol for real-time applications," RFC 1889, January 1996.
13. Casner, S. and Jacobson, V., "Compressing IP/UDP/RTP headers for low-speed serial links," Internet draft draft-ietf-avt-crtp-05.txt.
14. Hicks, M., Kakkar, P., Moore, J., Gunter, C., and Nettles, S., "PLAN: A packet language for active networks," *ICFP '98*, 1998.
15. Wetherall, D., Guttag, J., and Tennenhouse, D., "ANTS: A toolkit for building and dynamically deploying network protocols," *IEEE OPENARCH '98*, 1998.
16. Gordon R., *Essential JNI: Java Native Interface*, Prentice Hall Professional Technical Reference, 1st ed., March 1998.

Implementation of an Active Congestion Control Scheme in Narrowband ATM Networks

Sachin Sheth, Joseph Evans, Amit Kulkarni and Gary Minden
Department of Electrical Engineering and Computer Science
University of Kansas, Lawrence, KS 66045

Abstract

Narrowband integrated audio/data networks require bandwidth management techniques to be implemented during congestion to provide the desired quality of service. Congestion control schemes are traditionally applied generically at congested nodes. This can be improved by using active networks that provide payload-specific processing of user data at intermediate nodes in the network. Furthermore, these congestion control programs are inserted dynamically into the switches of a narrowband ATM network using on-demand loading. The challenges presented by a narrowband environment dictated some clever techniques to be employed to facilitate active networking. The effectiveness of this design is demonstrated by implementing a bit-dropping congestion control algorithm for audio coded using Sinusoidal Transfer Coding (STC).

1. Introduction

Traffic integration in communication networks is currently the focus of research activity in the telecommunication and computer communications communities. Standards have evolved to support high-bandwidth services over high-speed links. While these technologies are based on increasing bandwidth availability due to the fiber revolution, parallel developments in the fields of cellular and wireless communications must contend with rapidly depleting link capacities due to spectrum congestion. These developments underline the need for the implementation of integrated networks in a narrowband environment.

Since bandwidth is at premium in narrowband networks, congestion occurs frequently and control techniques have to be used to prevent undesired losses. However, applying a general congestion control algorithm to multimedia traffic is not the best solution. Different traffic types have different quality of service requirements, which demands traffic-specific processing. But it is not feasible to provide customized schemes for all traffic types on all switches in a practical network, considering the scale of today's networks.

An effective alternative is to implement the congestion control algorithms as application programs using active networking [1]. Active networking enables applications to dynamically extend the functionality of the network by injecting customized, application specific protocols, into the nodes and switches. In this type of network, packets select the protocol for processing from a list of protocols, which grows dynamically depending on the protocol support requirements of the connections. Nodes within this network become execution environments that supply Application Programming Interface for the protocols.

Many research groups have demonstrated the power and utility of active networking through the implementation of new services and protocols [7] [14] [15]. These implementations require packets to carry the protocol code and data along with them, which increases the size of the packets and thus increases transmission time. Additionally, they utilize an interpreted environment that can significantly slow down execution of the protocol code. The performance of such techniques for real-time traffic over wireless links has not been investigated yet. This scenario presents challenges with respect to efficiently transporting, installing and executing the protocol code at the network nodes over the wireless links to deliver adequate performance for real-time traffic. We address the problem of transporting and installing the code at the network nodes through the use of an out-of-band transport mechanism. Real-time execution performance is delivered through the use of code objects that contain implementations of the protocols in platform-specific binary code.

The outline of the paper is as follows. The next section discusses related work in the area of active networking. The section on implementation describes details of the design and implementation of the active network, software switches and the congestion control programs used for demonstrating the effectiveness of the proposed architecture. In the evaluation section, we discuss performance measurements carried out to evaluate the network and the results obtained. The paper ends with some conclusions and a list of references used.

2. Background and Motivation

Carrying mixed real time (audio) and non-real time (data) traffic over low speed, wireless links using standard ATM protocol presents many problems. The two principal problems are the high error rates due to channel fading, and the transmission latency of the cells. The modifications made to improve performance resulted in a Narrowband ATM [2] or NATM protocol. NATM uses a cell length of 24 payload bytes and 2 header bytes, as opposed to the standard 53-byte cell. Additionally, it utilizes a geometric Hamming Code based Forward Error Correction scheme that adds 50% redundancy to each cell. The smaller cell size reduces the transmission latency to acceptable levels for real-time traffic, and the FEC helps reduce errors to acceptable levels without retransmissions bringing down throughput.

Real-time traffic like audio is delay-sensitive as well as jitter-sensitive. The quality of service for such traffic is usually obtained in ATM networks by treating it as Constant Bit Rate (CBR) traffic and reserving resources for it. But the quality comes at a price. Assume the case of a low bandwidth connection being used for audio as well as data communication. Assigning a significant chunk of the already scarce bandwidth of the wireless link to the audio traffic to ensure zero drop-off in sound quality results in lower throughput for the data connection, which is typically bursty and lasts for a short interval. Audio data lends itself to compression and transcoding with/without source interaction. This suggests that audio traffic can be sent at a lower rate (lower bits per sample) by some method of bandwidth reduction for the duration of the data transfer. If the remaining bandwidth can be assigned to the data connection, then the cumulative performance of the wireless link is likely to be higher for the integrated traffic.

Examples of active processing can be found in existing congestion control mechanisms. Recognizing the performance degradation caused by fragmentation [3], packet level discard techniques have been explored to improve the performance of TCP/IP over ATM [4] [5]. The Partial Packet Discard strategy drops all cells in a packet that follow a dropped cell [4]; the Early Packet Discard strategy improves performance by aggressively dropping entire packets when a congestion threshold is reached [5]. The presence of a cell loss priority bit in the ATM cell header allows the source to indicate to the network that some cells should be treated differently than others under congestion.

Active networking has been applied to tackle the problems of congestion. Researchers at Georgia Institute of Technology [6] have targeted congestion control in ATM networks using active networking. They follow a functional paradigm in which packets select from a pre-defined set of active networking functions that have been pre-programmed into ATM switches in the network. While this paradigm provides more control over the packet's execution, it restricts dynamic customization of the network because it is impossible for the network to anticipate all possible requirements of user applications.

In a Turing machine model of an active network, nodes are programmable elements that allow applications to execute user-defined programs at the nodes. Therefore, nodes in an active network are called **active nodes**. Active networks exchange ***SmartPackets***, which contain a destination address, user data, and methods that are executed locally at any node in the active network. Active nodes perform the functions of receiving, scheduling, executing, monitoring and forwarding SmartPackets. New services and protocols are injected into the network by sending the protocol code within SmartPackets. Active nodes install the protocols by executing the SmartPackets.

An implementation of a selective filtering policy for wireless networks using the above approach has been discussed in [7]. In that effort, a terrain visualization application running over a wireless link panned the terrain by attempting to download image packets from a remote server. If congestion occurred, a filter installed at the wireless gateway selectively dropped packets that were outside the application's current view. However, transcoding delay-sensitive traffic such as audio in real-time and over wireless links presents many challenges. In an ATM network, the switches can execute an audio transcoding algorithm when they detect data transmission to reduce the bandwidth required by the audio traffic and provide better throughput to the data connection. To combat the bandwidth limitations of the wireless links, we use an out-of-band technique to install the requisite algorithms at the switches. This approach differs from the functional model used in [6]. Secondly, platform-specific binary code is used to implement the transcoding algorithm. This is unlike the approach used in [7], where the code object is in Java [8] bytecode format and the code is run in an interpreted environment. We use the binary code approach to achieve real-time performance. To implement this scheme in an active networking environment, we use the prototype active network toolkit called Magician [9], developed at the University of Kansas. In the next section, we describe an active congestion control scheme to handle real-time audio transcoding and the modifications made to the Magician toolkit to adapt it to deliver real-time performance over a Narrowband ATM network.

3. Active Congestion Control

ATM switches are usually built using custom hardware, since they are highly parallel and speed intensive applications. But in order to implement active functionality on an ATM switch, we choose to implement the switching capabilities in software so that the active processing can be integrated with the switching process. This provides a cost-effective solution for Narrowband ATM networks, without degradation in performance while utilizing the unused CPU cycles leftover during active processing by the active node manager The implementation of the software switch and its integration with the active node and the ATM subsystem is shown in Figure 1. The Q-port Signaler shown in the figure provides signaling stacks for a configurable number of ATM interfaces. The active manager is the Magician platform on the switch for providing active networking execution environment. ATM cells are switched by the MicroSwitch driver when no active processing is required i.e. no congestion control is required. However, when active processing is required, ATM cells are passed to the user space to be processed by the active manager before being switched out.

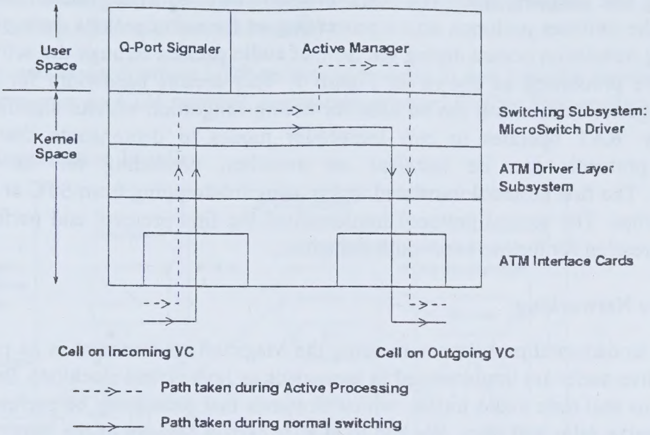

User Space
Kernel Space
Q-Port Signaler
Active Manager
Switching Subsystem: MicroSwitch Driver
ATM Driver Layer Subsystem
ATM Interface Cards
Cell on Incoming VC
Cell on Outgoing VC
Path taken during Active Processing
Path taken during normal switching

Figure 1. Block diagram of Software ATM Switch

To provide real-time performance on the low bandwidth links, we require a low bit-rate speech coder to encode the audio stream. Also, it is required that the speech coder support source-independent rate translation capability wherein it is possible to transform the bit rate of the coded speech stream at any intermediate node in the network, without interaction with the source. This feature is used in our congestion control scheme wherein the active nodes inside the network transcode the audio on congested links to lower bit rates to ease congestion. Embedded coders provide source-independent rate translation capability, where the parameters of the lower rate coder are embedded in the parameters of the higher rate coder [10]. To reduce the bit rate, the higher rate parameters are stripped off. The Sinusoidal Transfer Coder (STC) [11] allows inter-operability at different rates and source-independent rate translation using bit-dropping technique. STC is a multi-rate coder that operates at several discrete rates from 8 kbps to 2.4 kbps. In the current implementation, STC normally operates at 4.8 kbps. During congestion, STC performs bit-dropping and produces voice coded at 2.4 kbps. The data stream is not encoded since it is not be subject to active processing. Under congestion, data cells are dropped in the switch and transport mechanisms ensure reliability of data transfer.

Audio is encoded and encapsulated using Robust Audio Tool (RAT). The RAT is a network audio tool that allows users to participate in audio conferences over the Internet. Audio connections can be directly between two participants, or between a group of participants on a common multicast group. The RAT, as per IETF standards, uses RTP [12] over UDP/IP as its transport protocol, and conforms to the RTP profile for audio and video conferencing with minimal control. We implement a portion of the Internet draft [13], aimed towards compressing IP/UDP/RTP headers. In an ATM network, only RTP headers have to be compressed. Compression is done on a link-by-link basis as it provides better performance compared to an end-to-end basis, in terms of delay and loss rate. The design provides simplicity at the expense of generality in order to reduce the compression-decompression time.

When the RAT is operated in the transcoder mode it allows bandwidth translation between the incoming and outgoing link. The RAT program behaves as a protocol, which when installed on the switches performs active processing on the audio packets during congestion. Audio coding translation occurs during the flow of audio packets through the active manager through active processing as shown in Figure 1. This creates bandwidth for use on the outgoing congested link, which can be used for easing congestion without significant loss in audio quality. RAT operates in two transcoder modes to demonstrate that arbitrarily customized protocols can be installed on switches, depending on the connection requirements. The first protocol translated audio using bit-dropping from STC at 4.8 kbps to STC at 2.4 kbps. The second protocol implemented the first protocol and performed RTP header-compression for further bandwidth reduction.

3.1 Active Networking

One of the fundamental problems with using the Magician environment in its present form is that the active nodes are implemented in user space as Java virtual machines. But the target traffic contains real-time audio traffic, which demands that processing be performed at line rates to minimize delay and jitter. We had to modify certain features of the Magician code to integrate it with the software ATM switch. Here we discuss some of the modifications.

There are two major approaches towards transporting code in an active network, the out-of-band and the in-band approach. Magician uses the in-band approach in which the SmartPackets carry the protocol code as well as the data as they travel through the network. Network nodes process the packet by executing the accompanying code. In the out-of-band or discrete approach, the active nodes load and install the application-specific protocols during connection setup time. Data packets passing through these switches request processing by invoking one of the previously installed protocols. The Narrowband ATM Network uses low-bandwidth links to interconnect hosts. The size of the program code is significant enough to cause a long delay in transfer of the SmartPackets carrying audio data, if the in-band approach is used. Instead, we use an out-of-band loading scheme because that eliminates the latency of transporting and installing the code. The congestion control schemes are downloaded on to ATM switches in the connection path at connection set-up time.

The role of SmartPackets is different in this implementation. SmartPackets are used *only* to load program code in the active node. Since the out-of-band approach is used to load the program code using SmartPackets, these SmartPackets are sent out before the actual audio traffic is sent out. Each node on receiving the program code stores it locally in a cache from which it is retrieved when necessary. The program code will never be re-transmitted between same pair of interfaces, since each active node keeps track of the adjacent nodes that has received the particular program code previously.

The role of program code is different here as well. Program code, by itself, does not implement new protocols to be used by user-data. Instead, program code when executed causes execution of system-dependent binary code outside the Java virtual machine and the relevant parameters are passed to the binary code. The binary code in this case is used to implement congestion control algorithm. Thus, once SmartPackets install program code in the active node and the program is executed, the binary code is invoked. Other modifications made were:

- Magician identifies active nodes using IP host address and UDP port combination. In the Narrowband ATM Network, active nodes reside on software switches and are therefore identified using ATM interface addresses.
- Magician uses connectionless UDP sockets to transfer to and from the network. To make connections over an ATM network, connection-oriented PVC or SVC sockets are used.

3.2 Implementation Details

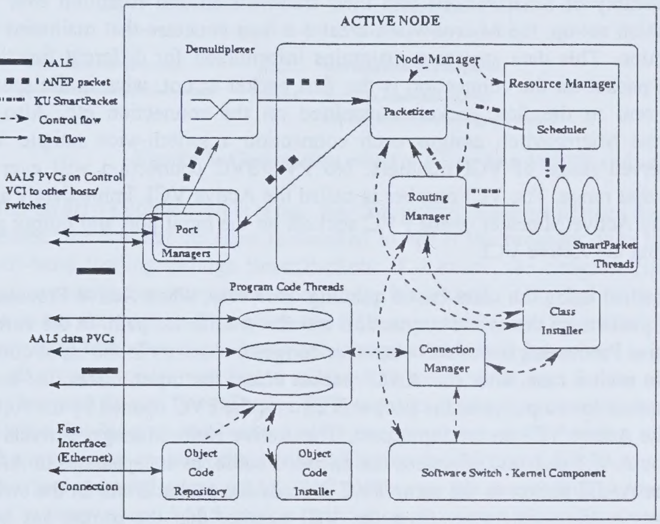

ACTIVE NODE
AAL5
ANEP packet
KU SmartPacket
Control flow
Data flow
Demultiplexer
Node Manager
Packetizer
Resource Manager
Scheduler
Routing Manager
SmartPacket Threads
AAL5 PVCs on Control VCI to other hosts/ switches
Port Managers
Program Code Threads
AAL5 data PVCs
Class Installer
Connection Manager
Fast Interface (Ethernet) Connection
Object Code Repository
Object Code Installer
Software Switch - Kernel Component

Figure 2. Active node architecture

The model of an active node in the Narrowband ATM network is shown in Figure 2. An active node is made up of (possibly) multiple processing environments such as a PLAN [14] execution environment or an ANTS [15] environment. For simplicity of demonstration, we assume that an active node contains only one execution environment. When an active node boots up, it starts a set of Port Managers, a Demultiplexer, and a Node Manager. There is one Port Manager per ATM interface on the end-host/switch. Usually, end-hosts have only one ATM interface, and hence only one Port Manager is started. A switch has one Port Manager for each enabled ATM interface.

The list of interfaces is obtained by querying the MicroSwitch through *ioctl()* calls using Java Native Interface (JNI). Since no information about the switch is available through standard system calls and Java does not provide support through its libraries, a library had to be created. Native functions were implemented which query the switch for different bits of information using *ioctl()* calls and return information to the Active Manager using Java Native Interface [16]. The Port Manager opens a PVC socket on a unique VCI (62) called Control VCI on each of the enabled interfaces. Since Java does not provide socket API for ATM, an API was defined and implemented which allowed AAL5 PVC socket management. The Port Managers read and write to these sockets and thereby communicate with immediate switch/host that is connected to the interface they are working with. The Port Manager on the corresponding host/switch listen on a socket opened on the same VCI. Incoming SmartPackets are queued at the Demultiplexer that verifies that the packet conforms to ANEP standard and then demultiplexes it to the correct processing environment. The byte stream constituting the packet is then handed over to the environment's Node Manager.

In the current implementation, the Node Manager starts other managers such as the Resource Manager, the Port Interface Manager, and the Connection Manager. In this implementation, there is no Routing Manager. The Port Interface Manager controls the assembly and disassembly of SmartPackets and their transmission and reception over the wire. During connection set-up, the MicroSwitch creates a data structure that maintains the state for the connection. This data structure maintains information for different functions such as whether the packet on the connection is the first packet or not, what is the special identifier that was sent in the first packet transmitted on the connection etc. Also, at connection set-up, the MicroSwitch assigns each connection a switch-wide unique VCI number from a reserved range of VCI numbers. No PVC/SVC connection will ever be assigned VCIs from that range. The VCI number is called the Active VCI. Immediately after connection set-up, the Active Manager opens PVC sockets on the input port and output port of the connection using the Active VCI.

Data is usually switched using the class-based queuing. However, when Active Processing is desired the AAL5 packets on the active connection traverse a different path. In the current implementation, Active Processing is desired whenever congestion occurs in the input queues for an output port. In such a case, after the AAL5 packet enters the input, instead of being queued in the input queue for output port, the packet is sent on the PVC opened by the Active Node Manager on the Active VCI on the input port. The Active Node Manager extracts the RTP packets from the AAL5 packets, processes the encoded audio, re-encapsulates in AAL5 format and writes the AAL5 packet to the same PVC it reads the packet from. In the switch kernel, the MicroSwitch driver interprets from the VCI number that the packet has been processed by the Active Node Manager and then queues up the packet on the input queue for the destined output port it is to be switched to. The path that the AAL5 packet then follows is the same that it would follow if there were no Active Processing. Figure 3 shows the diagrammatic implementation of the switching scheme.

The Connection Manager is responsible for checking creation of connections to the MicroSwitch and invoking binary code programs that implement congestion control processing on the user-data read through AAL5 PVCs. The Active Node Manager starts the Connection Manager. The Connection Manager polls the switch using native functions to determine if an active connection is set-up. Since active connections follow RTP standard, they always occur in pairs, one connection for the RTP data part and the other for the RTCP control part. Each active connection immediately after connection set-up stores a special identifier in every MicroSwitch in its connection path. This identifier in addition to the magic string "AN" contains the name of the class whose program code it wishes to execute on the connection. In this way the connection has control over what algorithm it wishes to implement when congestion occurs. Whenever an active connection is set-up, as mentioned an active VCI is also assigned to the connection. This is the VCI on which AAL5 PVC socket is opened by the binary code and is the VCI on which on which data is sent for active processing in the MicroSwitch before being switched out.

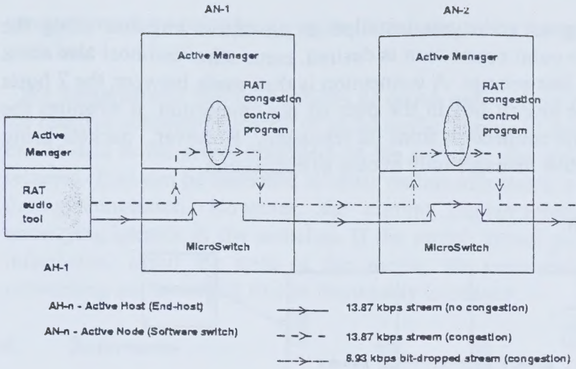

AN-1
AN-2
Active Manager
RAT
congestion
control
program
Active Manager
RAT
congestion
control
program
Active
Manager
RAT
audio
tool
Active
Manager
RAT
audio
tool
MicroSwitch
MicroSwitch
AH-1
AH-2
AH-n - Active Host (End-host)
AN-n - Active Node (Software switch)
13.87 kbps stream (no congestion)
13.87 kbps stream (congestion)
6.93 kbps bit-dropped stream (congestion)

Figure 3. Congestion control using bandwidth reduction

After an active connection is setup, the Connection Manager derives the program code to be executed. An internal database is checked to see if the program code has been installed by out-of-band loading through SmartPackets. If it exists, the program code is executed. The program code in turn attempts to execute the binary code it is required to invoke. If the binary code is not resident on the switch, an FTP request is made via a fast interface to fetch the binary code and store it on the switch. Program monitors have been implemented to ensure that all program codes wanting to execute the binary code wait for the single FTP request to complete and do not fire off more FTP requests. Once it is ensured that the binary code is available, the binary code is invoked by the program code by passing some arguments. One of the arguments contains the Active VCIs for the RTP and RTCP connections. The binary code when executed opens AAL5 PVC sockets on the Active VCI on the relevant interfaces. Thus, whenever an AAL5 packet arrives on the input side of an active connection, the MicroSwitch driver pushes it into the receive queue of the Active VCI socket for the active connection. The binary code reads the user-data from the AAL5 packet, processes it, re-encapsulates the data and sends it out on the socket. When the switch kernel receives data on an Active VCI socket, it is diverted to the MicroSwitch for switching out. Whenever an Active Connection is torn down, it is detected in the next Connection Manager poll and the Connection Manager stops the execution of the program code and binary code assigned to that connection.

3.3 Operation

During network setup, the active managers are initiated on all the switches. Depending on whether the machine is an end-host or a switch it starts the required managers e.g. end-hosts do not start the Connection Manager due to its obvious functionality. It is assumed that before active managers start up on end-hosts, all the user-specific protocols implemented using program code have been loaded appropriately. After the active manager starts up on the end-hosts, the user is prompted with a command-line prompt. The user keys in a command that includes the name of the class of the program code. This results in two things. First, the RAT application in started on that end-host. Second, the active manager sends the program code out on the PVC socket opened on a unique control VCI (62). The switch at the other end of the link receives the program code and stores it in its repository. It then sends the program code on the control VCI PVCs of all interfaces that are enabled on the switch. This results in a cascading effect and the program code gets installed in the active switches along the connection path. Since a point-to-point connection is desired, some other end-host also starts up the RAT using the command line prompt. A connection is then made between the 2 hosts desiring connection. If an active switch lies in the path of the connection, it executes the program code associated with the connection from its repository. However, packets being switched will be routed to the active manager only in case of congestion.

4. Performance Evaluation

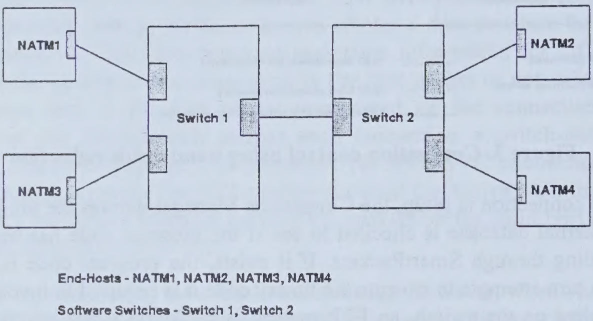

NATM1
NATM2
Switch 1
Switch 2
NATM3
NATM4
End-Hosts - NATM1, NATM2, NATM3, NATM4
Software Switches - Switch 1, Switch 2

Figure 4. Network topology

The efficacy of the scheme discussed above was evaluated using STC-coded audio. The network topology shown in Figure 4 was used to test the active networking functionality. The audio source initially produces 4.8 kbps coded audio. Two congestion control algorithms were tested. The first one was RAT in bit-dropping mode wherein the bandwidth of STC encoded audio was reduced from 4.8 kbps to 2.4 kbps. The second one was RAT in bit-dropping mode along with header compression to reduce the bandwidth on the congested link. Standard audio clips were sent from the transmitting end-hosts using RAT. Congestion was simulated on the switches causing active processing to be performed on the audio traffic. Audio clips were recorded on the receiving end-hosts. The performance of the clips using the different congestion algorithms was compared subjectively. It was observed that there was no perceivable drop-off in audio signal quality when congestion occurred and the audio traffic bandwidth was switched from 4.8 kbps to 2.4 kbps using the bit-dropping algorithm. When congestion was alleviated, the bit-dropping algorithm was not invoked and the audio bandwidth switched back to 4.8 kbps.

5. Conclusions

In this implementation it has been shown that an architecture which allows user-data to be processed within the switches in ATM networks, increases flexibility in protocol implementation within the network. An application packet has the freedom to select computation that occurs on its data as it flows through the network. This is especially useful in the current networks where integrated traffic is commonplace. Active networks enable customized protocols to be easily implemented. The use of the Java language to implement the active network makes it easy to deploy it on any computing system. The connection-oriented nature of ATM traffic allows the architecture to be scalable, as each interface needs to know only about the interface connected to the other end of the link. The time required for connection set-up provides an appropriate opportunity for pre-installing user protocols on the switches.

We demonstrated the applicability of active networks in narrowband networks by using it for reactive congestion control of audio traffic. Inspite of the additional processing performed on the data in the active managers on the switches, the real-time nature of audio traffic was retained. This can be extended to other multimedia traffic such as compressing digital video during congestion. However, the current implementation requires support from the underlying kernels in the switches. If the switch kernel provides a standard API that gives information about the state of the switch, the computational flexibility of using active networking on the switch can be immensely increased.

6. References

1. Tennenhouse, D. and Wetherall, D., "Towards an active network architecture," *Multimedia Computing and Networking*, January 1996.
2. Lindsley, E, *Narrowband ATM Networks*, M.S. Thesis, University of Kansas, 1997.
3. Kent C. and Mogul, J., "Fragmentation considered harmful," *ACM SIGCOMM*, 1988.
4. Armitage, G. and Adams, K., "Packet reassembly during cell loss," *IEEE Network Magazine*, September 1993.
5. Romanov, A. and Floyd, S., "Dynamics of TCP Traffic over ATM Networks," *IEEE Journal on Selected Areas of Communications*, May 1995.
6. Bhattacharjee, S., Calvert, K., and Zegura, E., "On active networking and congestion," *Technical Report GIT-CC-96/02*, Georgia Institute of Technology, 1996.
7. Kulkarni, A., and Minden, G., "Active networking services for wired/wireless networks," *IEEE INFOCOM '99*, 1999.
8. Gosling, J., and McGilton, H., *The Java Language Environment: A White Paper*, Sun Microsystems, 1995.
9. Kulkarni, A., Minden, G., Hill, R., Wijata, Y., Wahhab, F., Sheth, S., Gopinath, A., Nagarjan, A., and Pindi, H., "Implementation of an active network prototype," *IEEE OPENARCH '98*, 1998.
10. Champion, T., and Evans, J., "A flexible multirate speech coder," *Proc. Intl. Conf. On Signal Processing*, pp. 1440-1443, September 1993.
11. McAulay, R., and Quatieri, T., "Multirate sinusiodal transform coding at rate from 2.4 kbps to 8 kbps," *IEEE Intl. Conf. Acoustics, Speech, Signal Processing*, pp. 1645-1648, 1986.
12. Schulzrinne, H., Casner, S., Frederick, R., and Jacobson, V., "RTP: A transport protocol for real-time applications," RFC 1889, January 1996.
13. Casner, S. and Jacobson, V., "Compressing IP/UDP/RTP headers for low-speed serial links," Internet draft draft-ietf-avt-crtp-05.txt.
14. Hicks, M., Kakkar, P., Moore, J., Gunter, C., and Nettles, S., "PLAN: A packet language for active networks," *ICFP '98*, 1998.
15. Wetherall, D., Guttag, J., and Tennenhouse, D., "ANTS: A toolkit for building and dynamically deploying network protocols," *IEEE OPENARCH '98*, 1998.
16. Gordon R., *Essential JNI: Java Native Interface*, Prentice Hall Professional Technical Reference, 1st ed., March 1998.

CPU Scheduling for Active Processing using Feedback Deficit Round Robin

Tilman Wolf and Dan Decasper
{wolf, dan}@arl.wustl.edu
Applied Research Laboratory
Washington University
St. Louis, Missouri 63130

1 Introduction

To address the increasing demand for customized and application-specific processing on network routers, "active networks" have been proposed [1]. Active network routers are capable of processing code, which is either carried as part of the datagram's payload or referenced by a specific value in the datagram. Execution Environments (EEs) provide the context in which active processing is performed.

Process and thread scheduling for Execution Environments differs from regular time-slice scheduling as implemented by modern general-purpose operating systems. Active packets can contain individual code, which leads to context switches (additional to the context switches caused by scheduling) when packets with different code are processed in EEs. Contrary to general-purpose OSs, where this happens only when new applications are launched, these additional context switches can occur frequently when many packets contain individual code.

Context switching times of several thousand cycles are typical for switching between different processes [2]. This represents a considerable overhead in active networks, where CPU resources are already scarce. To reduce unnecessary context-switches, we propose Feedback Deficit Round Robin (FDRR) for CPU scheduling, which stems from Deficit Round Robin (DRR) [3], a fair-queuing scheme for packet scheduling.

2 Feedback Deficit Round Robin

FDRR uses the individual *expected processing time* as a criteria for CPU cycle management similar to the packet length used by DRR for bandwidth management. Since the individual processing time cannot be known ahead of time, an estimate is used. A feedback mechanism applies the measured processing time to adjust the estimate.

The outline of FDRR is shown in Figure 1. Each Execution Environment has a queue where packets are stored for processing. For each queue, a deficit counter and an estimate is maintained. The deficit represents the amount of processing that a particular queue can use. The estimate represents the expected processing time for the next packet.

The DRR scheduler forwards packets of a queue to the processor as long as the deficit is larger than the estimate of the next packet. With each packet, a timer is started that interrupts the processor in case a packet exceeds its resources. When the processing is finished or terminated, the actual processing time is used to adjust the deficit, as well as the estimate that is used for the next packet.

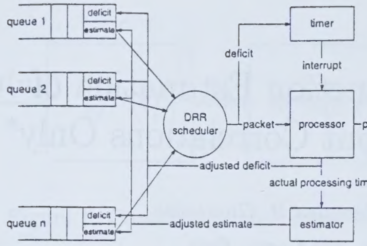


Figure 1: Feedback Deficit Round Robin

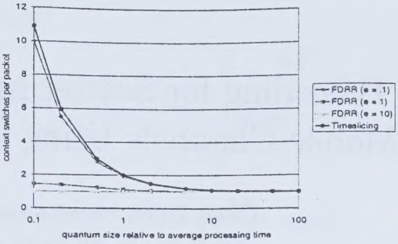


Figure 2: Number of Context Switches

3 Simulation Results

We assume for our measurements that each active packet can carry individual code. This means that a 'context switch' happens every time the processing of a new packet is started, because the instruction cache will not have the instructions cached, and state information from earlier packets has to be made available.

To compare the performance of FDRR with the timesliced scheduling, we simulated both algorithms over a range of parameters. The number of context switches per packet are shown in Figure 2. The quantum / timeslice ranges from .1 times to 100 times the average processing time of a packet. FDRR is simulated using three different estimates (.1, 1, and 10 times the average actual processing time).

Timeslicing causes most context switches for any quantum size. If FDRR underestimates the processing ($e = .1$) significantly, it performs similar to Timeslicing. If the estimation is the actual processing time ($e = 1$), FDRR incurs significantly fewer context switches for quantum sizes in the range of the actual processing time. FDRR causes the least number of context switches for large overestimations ($e = 10$). In this case, the packet is practically always processed to completion, and no interrupts or context switches due to scheduling occur. This case comes closest to the ideal number of context switches per packet of 1. For larger quantum sizes, many packets are being processed in the same round, and scheduling differences between FDRR and Timeslicing disappear.

Although FDRR with $e = 10$ seems to perform best, it also incurs most delay. While in Timeslicing a packet is processed immediately when it reaches the head of the queue, in FDRR the deficit is accumulated until it reaches the estimated processing time. This delay is significant for large overestimations. Thus, FDRR with a good estimation ($e = 1$) has the best overall performance.

References

[1] Tennenhouse, D., Smith, J., Sincoskie, D., Wetherall, D. [1997]. "A Survey of Active Network Research," *IEEE Communications*, 35, 1, January 1997, 80-86.

[2] Padmanabhan, V., Roselli, D. [1994]. "The Real Cost of Context Switching," *http://http.cs.berkeley.edu/~padmanab/class_projects/cs252.ps.*

[3] Shreedhar, M., Varghese, G. [1995]. "Efficient Fair Queuing using Deficit Round Robin," *Proc. of SIGCOMM 95*, ACM, Cambridge, Mass.

Precoding for Scattering Function Estimation of Mobile Channels Using Output Correlations Only*

Cihan Tepedelenlioğlu and *Georgios B. Giannakis*
200 Union St. SE, Dept. of Elect. and Comp. Engr.,
University of Minnesota, Minneapolis, MN, 55455
e-mail: `{cihan,georgios}@ece.umn.edu`
Tel: (612) 626-7781, Fax: (612) 625-4583

Abstract

We develop methods for estimating the channel correlations or equivalently the scattering function of mobile channels from output only data utilizing the knowledge of transmitted signal statistics. The desired transmitted signal statistics are induced by filterbank precoding or periodic insertion of one known symbol into the information stream, which would not have been enough for estimating the time-varying channel realization. We also propose a scheme for estimating vehicle velocity from the correlation information of the mobile channel. Simulations illustrate our approaches.

1 Introduction

The time and frequency selective propagation channel in mobile communications can be modeled as a time varying linear filter of finite length. In order to perform reliable detection, the channel impulse response needs to be estimated and tracked. Reliable channel tracking requires knowledge of the channel statistics (see e.g., [4, 14]). Second-order channel statistics can also be used in evaluating such important parameters as the Doppler spread, caused by the relative motion of the transmitter and the receiver, multipath spread, and level crossing rates, all of which affect critically system performance.

Kailath in his seminal paper [8] established the measurability limits of time-varying (TV) channels in terms of the product Doppler spread times multipath spread. In [8], he also proposed an estimator for the scattering function which characterizes the second order statistics of the channel. Bello defined the concept of a wide-sense stationary uncorrelated scattering channel and proposed analog estimators for the multipath and Doppler spread [2, 3]. Nonparametric methods for estimating the scattering function were proposed in [12] and [5] using a continuous-time framework where the deterministic input to the channel was designed for scattering function estimation. The discrete ambiguity function was used in [1], where again the channel input was assumed known.

In this paper we provide estimators for the channel correlations or equivalently the scattering function, that use the sampled output data and utilize knowledge of input

---

*This research is supported by ARO grant No. DAAG55-98-1-0336

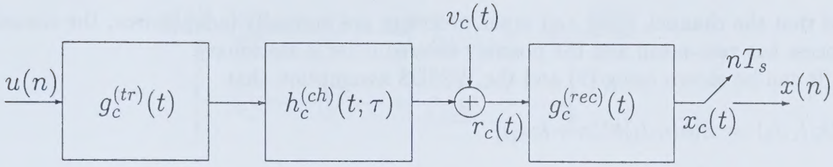


Figure 1: Continuous-time model for a baseband TV communication system

statistics that are induced using precoders, or periodic known symbol insertion within the transmitted block. The novel aspects of the paper include the estimation of channel correlations from the discrete time sampled output relying on transmitter induced channel statistics rather than knowledge of the input.

In Section 2, we will introduce the discrete-time formulation. Section 3 will describe the filterbank precoders that induce the desired input statistics for channel correlation estimation. In Section 4, known symbol insertion will be described for the purposes of correlation estimation especially for large lags which might be infeasible with filterbank precoding. In Section 5 we discuss a novel velocity estimation scheme where the moments of the Doppler spectrum are related to the channel correlations. Section 6 illustrates the approaches developed through simulations, and finally Section 7 concludes the paper.

2 Discrete-Time Formulation

Consider the block diagram of a communication system [1] in Figure 1. The relation between the channel input $u(n)$ and noisy channel output $r_c(t)$ can be written as,

$$r_c(t) = \int h_c(t;\tau)\left[\sum_k u(k) g_c^{(tr)}(t - kT_s - \tau)\right] d\tau + v_c(t) \;, \tag{1}$$

where $h_c(t;\tau)$ is the TV impulse response, $g_c^{(tr)}(t)$ and $g_c^{(rec)}(t)$ are the transmit and receive filter impulse responses respectively, and $v_c(t)$ is the noise. Assuming that the channel is approximately constant during a symbol duration T_s, ($h_c(t_1;\tau) \approx h_c(t_2;\tau)$, for $|t_1 - t_2| < T_s$), which is the case in most systems, the following holds:

$$x(n) = \sum_{l=0}^{L} h(n;l)u(n-l) + v(n) \;, \quad h(n;l) := \int h_c((n-l)T_s;\tau) r_g(lT_s - \tau) d\tau \;, \tag{2}$$

where L is the channel order, $r_g(t) := g_c^{(tr)} \star g_c^{(rec)}(t)$, $v(n) = v_c \star g_c^{(rec)}(t)|_{t=nT}$. We will use the input/output (I/O) relation in (2) throughout the paper in estimating the channel correlations defined by $E[h(n;l_1)h^*(n+k;l_2)]$. We will assume that the continuous-time TV channel $h_c(t;\tau)$ satisfies the wide sense stationary - uncorrelated scattering (WSS-US) condition [2],

$$E\left[h_c(t;\tau_1)h_c^*(t+\bar{t};\tau_2)\right] = \underbrace{r_c(\bar{t};\tau_1)}_{\text{WSS}}\underbrace{\delta(\tau_1 - \tau_2)}_{\text{US}} \;, \tag{3}$$

---

[1] subscript $_c$ denotes continuous time, $'$ for derivative, $\dagger$ for pseudoinverse, and $\star$ convolution

and that the channel, noise and input processes are mutually independent, the channel process has zero-mean and the possibly colored noise is stationary.

It can be shown using (2) and the WSSUS assumption that

$$r_h(k; l_1, l_2) := E[h(n; l_1)h^*(n+k; l_2)]$$

$$= \int E\left[h_c((n-l_1)T_s; \tau)h_c^*((n+k-l_2)T_s; \tau)\right] r_g(l_1T_s-\tau)r_g(l_2T_s-\tau)d\tau \ . \quad (4)$$

From (4), we see that despite the US assumption for the continuous time channel, $r_h(k; l_1, l_2)$ might not be zero for $l_1 \neq l_2$; a nonzero correlation between the different taps of the discrete-time channel could exist. But a closer examination of (4) will reveal that if the support of the transmit- and receive-filters are equal to the symbol duration T_s, then $r_h(k; l_1, l_2) = 0$ for $|l_1 - l_2| > 1$, i.e., only adjacent taps could be correlated. So, in what follows we will make the US assumption on the discrete-time equivalent channel as well, neglecting the adjacent tap correlation.

Using (2) we can express the received data correlation as:

$$r_x(n; k) = \sum_{l=0}^{L} r_h(k; l)\, r_u(n-l; k) + r_v(k) \ , \quad (5)$$

where $r_x(n; k) := E[x(n)x^*(n+k)]$, $r_h(k; l) := E[h(n; l)h^*(n+k; l)]$, and $r_u(n-l; k) := E[u(n-l)u^*(n-l+k)]$. Note that the autocorrelation function of $u(n)$ is allowed to depend on n. In fact, in what follows we argue that this is necessary for the estimation of $r_h(k; l)$. Given $r_u(n; k)$ and estimates of $r_x(n; k)$, we would like to estimate $r_h(k; l)$. It is immediate from (5) that if $u(n)$ is stationary, so that $r_u(n; k) \equiv r_u(k)$, then one can only hope for estimating $\sum_l r_h(k; l)$, rendering the correlation estimation for distinct taps impossible. Also apparent from (5) is that if $r_u(n; k) = 0 \ \ \forall n$, then $r_h(k; l)$ cannot be estimated. Hence, for a frequency selective, random fading channel, estimation of $r_h(k; l)$ requires $u(n)$ to be: (i)nonstationary, and (ii)have sufficient correlation extent.

Having delineated the basic requirements on $r_u(n; k)$ for the estimation of $r_h(k; l)$ to be feasible, we next describe how the discrete-time information sequence $s(n)$ can be precoded with a transmit filterbank to create a new sequence $u(n)$ with statistics that satisfy (i) and (ii).

3 Filterbank Precoding

Consider the filterbank precoder in Figure 2, where $s(n)$ is the sequence of information symbols, and $u(n)$ is the precoded channel input. It was shown in [13] that such a filterbank can be used as a discrete-time equivalent model for popular transmissions such as CDMA and OFDM with the proper selection of the $f_m(n)$ filters. The I/O relation of the precoder is given by

$$u(n) = \sum_{i=-\infty}^{\infty} \sum_{m=0}^{M-1} s(iM+m)f_m(n-iP) \ . \quad (6)$$

We will use $u(n)$ as input to the TV channel to estimate $r_h(k; l)$, which will enable not only estimation of channel correlations but also transmission of information symbols

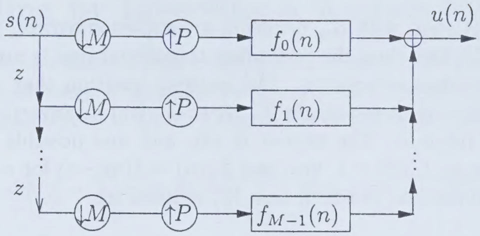


Figure 2: Discrete-Time Precoder Model

simultaneously, - a feature that no other method (to our knowledge) in the literature offers. Consider the received data correlation:

$$r_x(n;k) = \sum_{l_1,l_2} E[h(n;l_1)h^*(n+k;l_2)]$$

$$\times \sum_{i_1,i_2=-\infty}^{\infty} \sum_{m_1,m_2=0}^{M-1} E[s(i_1M+m_1)s^*(i_2M+m_2)] f_{m_1}(n-l_1-i_1P) f^*_{m_2}(n-l_2-i_2P+k) \ . \quad (7)$$

Assuming that $s(n)$ is white with unit variance, the US condition holds for $h(n;l)$, the filters are FIR of length P ($f_m(n) = 0$ for $n \notin [0, P-1]$ and $M > L$, we obtain

$$r_x(n;k) = \sum_{l=0}^{L} r_h(k;l) \left[\sum_{i=-\infty}^{\infty} \sum_{m=0}^{M-1} f_m(n-l-iP) f^*_m(n-l-iP+k) \right] \ . \quad (8)$$

We append L trailing zeros (TZ) to the filters $f_m(n)$, which restricts the support of $f_m(n)$ to $P-L$: $f_m(n) = 0$ for $n = P-L, \ldots, P-1$. The polyphase components of $r_x(n;k)$ can then be written as:

$$r_x(nP+p;k) = \sum_{l=0}^{L} r_h(k;l) \left[\sum_{m=0}^{M-1} f_m(p-l) f^*_m(p-l+k) \right] \ , \quad (9)$$

because, due to the L trailing zeros, only the $i = n$ summand contributes in (8).

We can also write (9) in matrix form as:

$$\mathbf{r}_x(k) = \mathbf{F}_2(k)\mathbf{r}_h(k) + \mathbf{1}_P r_v(k) \ , \quad (10)$$

where $[\mathbf{r}_x(k)]_p = r_{xx}(nP+p;k)$, $[\mathbf{r}_h(k)]_l = r_h(k;l)$, $[\mathbf{F}_2(k)]_{p,l} = \sum_{m=0}^{M-1} f_m(p-l) f^*_m(p-l+k)$, and the length P vector $\mathbf{1}_P := [1 \ldots 1]^T$.

Because $\mathbf{F}_2(k)$ is a tall ($P > L$) banded Toeplitz matrix, it will always be full rank as long as it has at least one nonzero entry.

Note that due to the finite support of the filters $f_m(n)$, $\mathbf{F}_2(k) = \mathbf{0}$ for $k > P-L$. Therefore, in order to estimate the channel correlations up to a lag k, we need to choose $P \geq k+L$. How should we choose $f_m(n)$ so as to make $\mathbf{F}_2(k)$ full rank for $k \in [0, P-L]$? Interestingly, OFDM precoders yield $\mathbf{F}_2(k) = \mathbf{0}$ for $k \neq iM$, where i is an integer and $M := P-L$. The reason is that the OFDM systems take the FFT of the sequence $s(n)$, which does not introduce color into $u(n)$.

Choosing $f_m(n) = \alpha_m \; \forall n$, with α_m denoting a complex constant, yields a full-rank $\mathbf{F}_2(k)$ for $k \in [0, P-L]$, but then the precoding transformation is not invertible since $f_m(n)$ are linearly dependent sequences. The natural question that arises is whether there exist linearly independent sequences $f_m(n)$ that yield an invertible $\mathbf{F}_2(k)$ for $k \in [0, P-L]$, the lags of interest. The answer is yes, and one possible example of such precoders can be chosen as $f_m(0) = 1, \forall m$, and $f_m(n) = \delta(m-n)$ for $n \neq 0$. This choice makes $\mathbf{F}_2(k)$ a diagonal matrix, in which case (8) reduces to:

$$r_x(nP+p;k) = r_h(k;p) + r_v(k) \; , \quad k \in [1, P-L] \; . \tag{11}$$

When $k = 0$, the first column of $\mathbf{F}_2(0)$ will be $[M, \mathbf{1}_{M-1}^T, \ldots, 0]^T$, and the first row is $[M, 0, \ldots, 0]$, which defines the entire matrix since $\mathbf{F}_2(k)$ is Toeplitz.

Such a choice for $f_m(n)$ enables the estimation of $r_h(k;l)$ from $r_x(nP+p;k)$ directly (c.f. (11)) for $k \neq 0$, and is computationally simple since it only requires a matrix multiplication with the (precomputed) $\mathbf{F}_2(0)^\dagger$ for $k=0$. In the presence of noise though, we can improve our estimates of $r_h(k;l)$ for any choice of $f_m(n)$ yielding a full rank $\mathbf{F}_2(k)$ for $k \in [0, P-L]$ as follows: Let $\mathbf{G}(k)$ be comprised of all but the last row of $[\mathbf{F}_2(k)|\mathbf{1}]^\dagger$. Then clearly $\mathbf{G}(k)\mathbf{F}_2(k) = \mathbf{I}$, and $\mathbf{G}(k)\mathbf{1} = \mathbf{0}$, which means that operating on both sides of (10) we obtain: $\mathbf{G}(k)\mathbf{r}_x(k) = \mathbf{r}_h(k)$. We have therefore nullified the affect of stationary and possibly colored noise once we have the ensemble values of the output statistics.

It is straightforward to generalize this approach when no trailing zeros are padded in the transmitted sequence, in which case we will have

$$[\mathbf{F}_2(k)]_{p,l} = \sum_{m=0}^{M-1} \left[f_m(p-l) f_m^*(p-l+k) + f_m(P+p-l) f_m^*(P+p-l+k) \right] \; , \tag{12}$$

but the relationship between $f_m(n)$ and $\text{rank}(\mathbf{F}_2(k))$ is not as transparent as the trailing zeros case. Nevertheless, for the block-transmission system at hand (for a fixed $f_m(n)$) it is easy to construct $\mathbf{F}_2(k)$ and check whether it has full-column rank.

Notice that when $M = 1$, $u(n) = \sum_i s(i) f(n-iP)$ is a spread spectrum signal where $f(n)$ is the spreading code. Using the techniques of this section, it is possible to estimate $r_h(k;l)$. In particular, the multipath intensity profile, $r_h(0;l) = E|h(n;l)|^2$ can be estimated. The quantity $r_h(0;l)$ is useful in providing the intensity and the locations of the multipath power measured in symbol durations, and could be important in assessing the amount of diversity offered by multipath to combat fading.

As mentioned previously, estimation of $r_h(k;l)$ requires the block length to satisfy $P \geq L + k$. In mobile communication systems, the scattering function $S_h(\omega;l) := \sum_k r_h(k;l) e^{-j\omega k}$ is always bandlimited for every l, and its bandwidth depends on the Doppler spread, which in turn depends linearly on the speed of the mobile through the carrier frequency. Due to the bandlimited nature of $S_h(\omega;l)$, the nonzero support of the correlations $r_h(k;l)$ in the lag domain k will be large, necessitating block lengths P on the order of several thousands. In the next section, we introduce a new method that inserts a known symbol every $K \gg L$ symbols of $s(n)$, enabling the estimation of $r_h(k;l)$ for large lags k.

4 Precoding by Inserting a Known Symbol

Suppose we obtain the channel input $u(n)$ from the information sequence $s(n)$ by inserting a known symbol every K symbols (which without loss of generality we will assume it to be a 1) so that $u(Kn) = 1 \; \forall n$. Then, for $m \neq 0$ and $K > L$ the following holds:

$$E\left[x(l_0)x^*(l_0+mK)\right] = \sum_{l=0}^{L} r_h(mK;l)\left[\sum_{i,j=-\infty}^{\infty} \delta(l_0-l-iK)\delta((m-j)K)\right] + r_v(mK)$$

$$= r_h(mK;l_0) + r_v(mK) \tag{13}$$

Using (13), we can estimate $r_h(k;l)$ for any tap $l = l_0$, and since the Doppler spectrum for each tap $S_h(\omega;l)$ is bandlimited, $r_h(mK;l_0)$ can be interpolated to find $r_h(k;l)$ for any k.

This approach can be extended to when $r_h(k;l_1,l_2) \neq 0$ for $l_1 \neq l_2$. In that case, it is straightforward to show that $E[x(l_1)x^*(l_1+mK+l_2)] = r_h(mK+l_2-l_1;l_1,l_2) + r_v(l_2+mK)$. Since $K \gg l_2 - l_1$, we can assume $r_h(mK+l_2-l_1;l_1,l_2) \approx r_h(mK;l_1,l_2)$ and interpolate $r_h(mK;l_1,l_2)$ to find $r_h(k;l_1,l_2)$ for any k, if needed.

The correlations of $x(n)$ in (13) can be estimated as usual using sample averaging:

$$\hat{r}_h(mK;l_0) = \frac{1}{N}\sum_{i=0}^{N-1} x(iK+l_0)x^*((i+m)K+l_0) \tag{14}$$

Assuming that the channel taps are Gaussian processes, we have found the variance of $\hat{r}_h(mK;l_0)$ to be

$$\mathrm{var}(\hat{r}_h(mK;l_0)) = C_1\mathcal{O}(\frac{L^2}{N}) + C_2\mathcal{O}(\frac{L}{N}) + C_3\mathcal{O}(\frac{1}{N}) + C_4\mathcal{O}(N^{-2p}) \tag{15}$$

where p is such that $|r_h(k;l_0)| \to 0$ as k^{-p}, and C_i's are constants independent of N. The last term in (15) is only related to the decay rate of $|r_h(k;l_0)|$, and is present even when $|r_h(k;l_0)|$ is estimated from the realizations of $h(n;l)$. For the correlation function corresponding to the Jakes's spectrum we have $r_h(k;l) = \sigma_h^2(l)J_0(\omega_D k)$, where $J_0(.)$ is the zero order Bessel function of the first kind, and ω_D is the normalized maximum Doppler frequency [7]. The value of p which dictates the decay rate in this case is $p = 1/2$.

5 Vehicle Velocity Estimation

Estimation of the mobile velocity is important for purposes of handoff. This information can be obtained through the level crossing rates of the envelope of the received signal [7], or alternatively, from the channel correlation [9]. We will pursue this latter approach and relate the Doppler spread (which is proportional to the velocity of the vehicle in the direction of the line joining the transmitter and the receiver) to the correlations of the channel taps.

Suppose for the sake of simplicity, the channel has one tap $h(n) = h_c(t)|_{t=nT}$ whose bandlimited spectrum is given by $S_h(\omega)$. We are interested in finding the bandwidth of $S_h(\omega)$ from the correlations $r(k) = E[h(n)h^*(n+k)]$. More precisely, we want to estimate the minimum ω that satisfies $S_h(\overline{\omega}) = 0$ for $|\overline{\omega}| > \omega$. The relation between

ω_D and the vehicle velocity v is given by $\omega_D = (2\pi v f_c)/(f_s c)$, where f_c is the carrier frequency, f_s is the symbol rate, and c is the speed of propagation.

One approach to estimating ω_D could be to find the bandwidth of $\hat{S}_h(\omega)$ corresponding to $\hat{r}_h(k)$. The problem with this approach is due to the difficulty of estimating $r_h(k)$ for large enough lags k because for practical bit rates this would amount to lags on the order of thousands.

An alternative given in [9] assumes that $r_h(k) = \sigma_h^2 J_0(\omega_D k)$ and upon approximating the Bessel function with a polynomial, it matches the polynomial (which depends on ω_D) to the estimated correlation $\hat{r}_h(k)$ in the least square sense. The algorithm relies on Newton's method to find the roots of a high-order polynomial whose degree depends on the approximation order of $J_0(\omega_D k)$.

In what follows we show that once a certain correlation function is assumed (such as $r_h(k) = \sigma_h^2 J_0(\omega_D k)$), the knowledge of $r_c(0) := E|h(t)|^2$ and $r_c''(0)$ are sufficient to estimate ω_D. It is well-known that the moments of the spectrum can be related to the derivative of the correlation function [11],

$$-\frac{r_c''(0)}{r_c(0)} = \frac{\int \omega^2 S_h(\omega) d\omega}{\int S_h(\omega) d\omega}\ . \tag{16}$$

The right hand side of (16) is a measure of Doppler spread and if $r_h(k) = \sigma_h^2 J_0(\omega_D k)$ corresponding to $S_h(\omega) = \sigma_h^2[\pi(\omega_D^2 - \omega^2)]^{-1/2}$, the relationship between ω_D and the correlations becomes

$$\omega_D = \sqrt{\frac{-2r_c''(0)}{r_c(0)}}\ . \tag{17}$$

Equation (10) is intuitive because if the curvature $r_c''(0)$ is bigger, the correlation function is broader, implying narrower bandwidth and smaller ω_D.

The quantities $r_c(0) = r_h(0)$ and $r_c''(0)$ can be estimated through the first few lags of $r_h(k)$ using finite differences. But since in practice, we will have $\hat{r}_h(\tau)$, estimating $r_c''(0)$ from $\hat{r}_h(\tau)$ using finite differences might be less reliable. An alternative could be fitting a parabola to $r_h(k) = ak^2 + b$ in the least square sense for $k = 0, \ldots, k_{max}$, and using $\hat{r}_c''(0) = a$ and $\hat{r}_c(0) = b$.

6 Simulations

In our simulations we adopted the carrier frequency and data rate of the North American Digital Cellular system which are 900 MHz and 24 KHz respectively. The temporal statistics of each channel tap are governed by the Jakes's model. In all figures SNR was set to 10 dB.

In Figure 3, we plot the multipath intensity profile $r_h(0, l)$ with respect to l. We depict the true $r_h(0, l)$, its estimate obtained from averaging the realization of the channel $h(n; l)$, and the estimate obtained through our filterbank precoding method with $M = 1$ (i.e., a spread spectrum signal as channel input). The mobile speed was $v = 100$ km/hr, which corresponds to a maximum Doppler spread of 83 Hz. A data record of length 12,000 was used. This corresponds to a time span of 0.5 seconds, which is a short enough time span for the statistics of the channel to remain unchanged.

In Figure 4 we depict the correlations of the first tap of a two-tap channel. The smooth line is the estimate obtained from the realizations of the channel, and the non-smooth line is the estimate obtained through the known symbol insertion method with $K = 10$, and each lag being $KT_s = 4.1 \times 10^{-4}$ seconds. As with Figure 3, in Figure 4, a data record corresponding to a 0.5 seconds of the output data was used, and the mobile speed was 100 km/hr.

In Figure 5 we depict the ratio of the estimated over the true velocity for different speeds. We see that the estimator has increasing bias for higher velocities, but for the velocity ranges of interest, it exhibits satisfactory performance. In Figure 5, $N = 4{,}800$ data were used, which corresponds to 0.2 seconds for a 24 KHz data rate transmission.

7 Conclusions

In this paper, we developed methods for estimating the autocorrelation sequence of the mobile channel. After determining the requirements on the statistics of the transmitted signal for channel correlation estimation, we proposed two methods that precode the white information bearing symbols in order to induce the desired statistics. Then, we related channel correlations to the Doppler spread, which in turn determined the mobile speed. Preliminary simulations illustrated satisfactory performance of the methods.

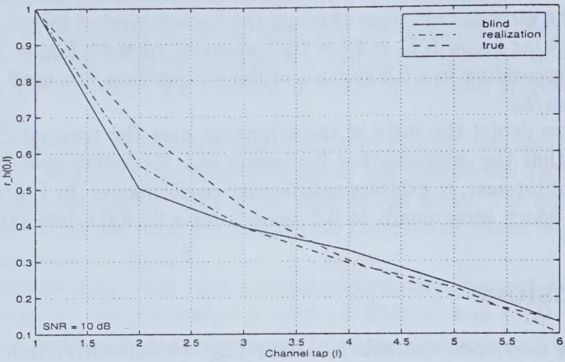


Figure 3: Multipath Intensity Profile

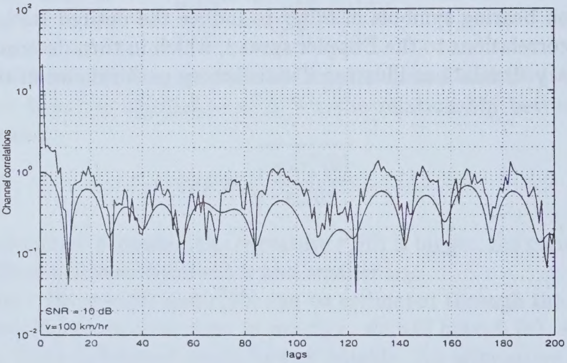


Figure 4: Channel correlations for one tap

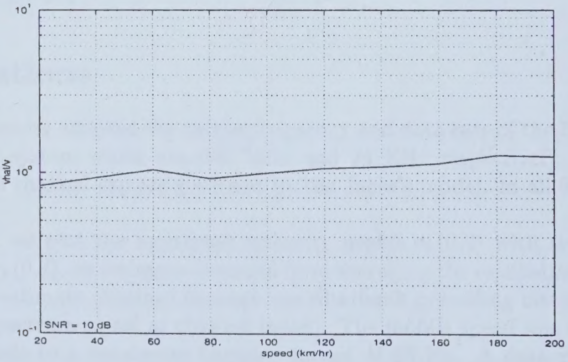


Figure 5: Estimation of speed for different velocities

References

[1] H. Artes, G. Matz, and F. Hlawatsch, "An Unbiased Scattering Function Estimator for Fast Time-Varying Channels", *Proceedings of SPAWC '99*, pp. 411-414 May 1999.

[2] P. A. Bello, "Characterization of Randomly Time-Variant Linear Channels," *IEEE Transactions on Communication Systems*, vol. CS-11, pp. 360-393, December 1963.

[3] P. A. Bello, "Some Techniques for the Instantaneous Real-Time Measurement of Multipath and Doppler Spread," *IEEE Transactions on Communication Technology*, vol. 13, No. 3, pp. 285-292, September 1965.

[4] S. A. Fechtel, H. Meyr, "Optimal Parametric Feedforward Estimation of Frequency-Selective Fading Radio Channels", *IEEE Transactions on Communications*, vol. 42, No. 2, pp. 1639-1650 February 1994.

[5] N. T. Gaarder, "Scattering Function Estimation", *IEEE Transactions on Information Theory*, vol. IT-14, No. 5, pp. 684-693, September 1968.

[6] J. M. Holtzman and A. Sampath, "Adaptive Averaging Methodology for Handoffs in Cellular Systems", *IEEE Transactions on Vehicular Technology*, vol. 44, no. 1, pp. 59-66 February 1995.

[7] W. C. Jakes, *Microwave Mobile Communications*, John Wiley & Sons, Inc., 1974.

[8] T. Kailath, "Measurements on Time-Variant Communication Channels", *IEEE Transactions on Information Theory*, vol. 8, pp. 229-236, September 1962.

[9] J. Lin, J. G. Proakis, "A Parametric Method for Doppler Spectrum Estimation in Mobile Radio Channels", *Proceedings of CISS*, pp. 875-880, March 1993.

[10] J. Lin, J. G. Proakis, "Optimal Tracking of Time-Varying Channels: A Frequency Domain Approach for Known and New Algorithms", *IEEE Transactions on Communications*, vol. 13, No. 1, pp. 141-153, January 1995.

[11] A. Papoulis, *Probability Random Variables and Stochastic Processes*, McGraw-Hill, 1984.

[12] D. W. Ricker, M. J. Gustafson, "A Low Sidelobe Technique for the Direct Measurement of Scattering Functions", *IEEE Transactions on Oceanic Engineering*, vol. 21, No. 1, pp. 14-23, January 1996.

[13] A. Scaglione, G. B. Giannakis, and S. Barbarossa, "Redundant Filterbank Precoders and Equalizers, Part II: Blind Channel Estimation, Synchronization, and Direct Equalization," *IEEE Transactions on Signal Processing*, vol. 47, pp. 2007-2022, July 1999.

[14] M. K. Tsatsanis, G. B. Giannakis, and G. Zhou, "Estimation and equalization of fading channels with random coefficients," *Signal Processing*, vol. 53, pp. 211-229, 1996.

Two-Stage Hybrid Acquisition of Multicarrier Direct-Sequence Spread-Spectrum Signals

Frederick J. Block and Carl W. Baum*
Department of Electrical and Computer Engineering
Clemson University
Clemson, SC 29634-0915

Abstract

The frequency diversity of multicarrier direct-sequence signaling can potentially offer robust performance in frequency-selective channels. This paper focuses on hybrid acquisition of multicarrier signals in channels containing fading. By taking advantage of the multiple correlators required for multicarrier reception, the receiver's acquisition stage can investigate multiple candidate delays simultaneously. A class of two-stage acquisition techniques are presented that are specifically designed for multicarrier direct-sequence signaling. The designs are optimized to minimize the resulting average acquisition time in fading channels.

1 Introduction

Multicarrier direct-sequence (DS) code division multiple access (CDMA) systems have been proposed as alternatives to single-carrier DS CDMA. Instead of transmitting a single wideband DS signal, several DS signals with smaller bandwidths are transmitted in parallel. Multicarrier CDMA provides good performance in multiple-access interference if families of spreading sequences are used that have good crosscorrelation properties. Multicarrier CDMA also provides protection against frequency-selective fading if the bandwidth of each signal is less than the coherence bandwidth of the channel [1]. Furthermore, multicarrier CDMA protects against partial-band interference [2] because of the frequency diversity it provides. If the number of carriers is allowed to vary, multicarrier CDMA can readily provide multiple data rates, and thus it is amenable to multimedia communication.

Before a multicarrier DS CDMA system can begin receiving data, the receiver must acquire the timing of the spreading sequences used. In selecting an acquisition scheme, there is a fundamental tradeoff between performance and complexity. At one extreme are serial acquisition schemes, which test one candidate delay at a time. Serial acquisition is slow, but low in complexity. At the other extreme are fully parallel acquisition schemes that test all candidate delays simultaneously. Parallel acquisition is fast, but high in complexity. Between the extremes are hybrid acquisition schemes which test an intermediate number of delays at one time [3].This paper focuses on the design of hybrid acquisition schemes *specifically* for multicarrier CDMA.

Among hybrid acquisition techniques are single-stage and two-stage schemes. With single-stage schemes, a group of candidate delays is tested at one time. If the test result for a particular delay meets some criteria such as exceeding a threshold, then that

---

*This work was supported in part by the National Science Foundation under grant number NCR-9523788.

candidate delay is assumed to be correct and acquisition stops. Otherwise, the receiver tests a new group of candidate delays.

With two-stage acquisition schemes, the acquisition process is divided into two parts: searching and verification [4]. The first stage is used to search quickly through groups of candidate delays and make decisions on which delays are most likely correct. Those delays whose first-stage test results indicate that they are likely to be correct are passed to the second or verification stage. The second stage is designed to accurately determine whether the candidate delay is correct, so generally a longer testing period is used than in the first stage. Since the longer verification stage is used for relatively few candidate delays, the two-stage scheme generally has a shorter average acquisition time than a single-stage scheme [4].

A receiver for an L carrier system employs at least L correlators, one for each subband. Having L correlators provides a great deal of flexibility for acquisition. The receiver can investigate many candidate delays at a time in the first stage by using a single correlator for each candidate. Alternatively, the receiver can process a much smaller number of candidate delays simultaneously by looking at all L subbands for each candidate delay. In addition, acquisition schemes between these two extremes can be considered. We consider each of these approaches.

There is additional flexibility in the allocation of correlators for the second-stage testing. Once processing in the first stage results in a decision that a particular candidate delay is likely to be correct, the receiver can then use anywhere from one to L correlators for verification. If the number of correlators used for verification is small, the remaining correlators can continue to perform first-stage testing of other candidate delays. Using a single correlator for verification has been explored previously for hybrid single-carrier DS acquisition [5]. In this paper, we consider both approaches with the goal of determining the scheme with the best performance.

The remainder of the paper is organized as follows. In Section 2, the channel and system models are given. In Section 3, the hybrid acquisition schemes are described. An approximation to the performance of one hybrid scheme is presented in Section 4. The performance results are given in Section 5.

2 Channel and System Models

The multicarrier system considered transmits DS signals on L carriers. The L associated subbands are assumed to be disjoint in frequency. The signal in each subband is assumed to be bandlimited so that no interference affects the signals in other subbands. The signal transmitted in the ith subband is given by $s_i(t) = \sqrt{2P}d(t)c_i(t)\cos(2\pi f_i t)$, where P is the transmitted power, $d(t)$ is the data waveform, $c_i(t)$ is the spreading signal for the ith subband, and f_i is the carrier frequency for the ith subband. The ith spreading signal is $c_i(t) = \sum_{j=-\infty}^{\infty} \hat{c}_i(j)p(t - jT_C)$, where $p(t)$ is the chip waveform, T_C is the time per chip, and $\{\hat{c}_i\}$ is the spreading sequence with $\hat{c}_i(j) \in \{-1, 1\}$. The bandwidth of $s_i(t)$ is denoted by W_M. The composite transmitted signal is $s(t) = \sum_{i=1}^{L} s_i(t)$, and the total signal bandwidth is $W = LW_M$. We do not require that the L subbands be contiguous.

The channel is assumed to contribute noise and exhibit frequency-selective fading. The noise is additive white Gaussian noise (AWGN) with power spectral density $N_0/2$. The signal in each subband is assumed to undergo frequency non-selective Rayleigh fading. (The fading for the composite signal, however, *is* frequency dependent.) The amplitudes of the L signals are modeled as independent, identically distributed Rayleigh random variables, and the L carrier phases are modeled as independent, identically distributed uniform random variables. This model of "flat," independently faded subbands is most accurate for systems in which the subbands are not contiguous. The delays of the L signals are assumed to vary from each other by an amount less than a chip duration. The fading is assumed to vary sufficiently slowly so that the amplitude and phase of each carrier is constant during acquisition.

The received signal is

$$r(t) = \sum_{i=1}^{L} \sqrt{2}\, d(t)\rho_i c_i(t-\tau_i)\cos(2\pi f_i t + \theta_i) + N(t),$$

where ρ_i is the received amplitude of the ith carrier, θ_i is the received phase, τ_i is the unknown delay, and $N(t)$ denotes the noise and interference. Each ρ_i is Rayleigh with second moment $2\beta^2 T_C^{-2}$, and each θ_i is uniform over $[0, 2\pi)$. The total average received energy per chip is given by $E_C = 2L\beta^2 T_C^{-1}$. For the numerical results in this paper, we model the τ_i's as equal to each other and equal to UT_C, where U is an integer uniformly distributed in $[1, M]$. We also model the spreading sequences on the L carriers as ideal. It is assumed that no other users are present and that no data is transmitted during acquisition (i.e. $d(t) = 1$).

3 Hybrid Acquisition Schemes

In this section, we describe the two-stage acquisition schemes under consideration. We assume that a total of L noncoherent correlators are available for acquisition, where L is also the number of carriers in the multicarrier CDMA signal. In the first stage, the receiver investigates K delays at a time. For each candidate delay, $J = L/K$ correlators are used, each tuned to a different carrier. (We assume that L/K is an integer.) Correlation takes place over n_1 chips. The correlator for the ith subband of the ℓth candidate delay produces two outputs, an in-phase component and a quadrature component. The in-phase component is denoted by X_i^ℓ and the quadrature component is denoted by Y_i^ℓ. The values for X_i^ℓ and Y_i^ℓ are determined by

$$X_i^\ell = \sqrt{2}\int_0^{n_1 T_C} r(t)c_i(t-\ell T_C)\cos(2\pi f_i t)\,dt \tag{1}$$

and

$$Y_i^\ell = \sqrt{2}\int_0^{n_1 T_C} r(t)c_i(t-\ell T_C)\sin(2\pi f_i t)\,dt. \tag{2}$$

The decision statistic for the ℓth candidate delay is

$$Z_\ell = \sum_{i=1}^{J} (X_{g(i)}^\ell)^2 + (Y_{g(i)}^\ell)^2, \tag{3}$$

where $g(i)$ represents the ith subband used in the decision statistic. Once the decision statistics for the K candidate delays are generated, the receiver finds the largest of the decision statistics and compares it to the threshold τ_1. If it exceeds the threshold, the corresponding delay is checked in the verification stage.

We consider two approaches with regards to verification. In the first approach, once a candidate delay is accepted by the first stage, all L correlators are then used for verification. In this way, statistics are obtained from all L subbands. The correlation time for verification is n_2 chips. Noncoherent correlation is used so that the statistics are similar to (1) and (2) except that n_1 is replaced by n_2. After correlation, the decision statistic

$$\hat{Z}_\ell = \sum_{i=1}^{L} (\hat{X}_i^\ell)^2 + (\hat{Y}_i^\ell)^2$$

is obtained, where $\hat{X}_i^\ell$ and $\hat{Y}_i^\ell$ are the in-phase and quadrature outputs. If $\hat{Z}_\ell$ is larger than the threshold τ_2, then the acquisition stage declares the delay to be correct. Otherwise, the system returns to the first stage and begins processing the next K delays. We refer to this approach as *batch verification*. This approach to two-stage hybrid acquisition has been studied for single-carrier CDMA in [3].

Instead of using all L correlators for verification, the receiver may use only a fraction of the available correlators to verify a potential delay. Here, the remaining correlators may continue searching through sets of candidate delays or, as appropriate, may verify different candidate delays. We refer to this approach as *parallel verification*. This approach was first considered for single-carrier acquisition in [5].

With parallel verification, we assume that J correlators are used per delay for both testing and verification. Because multiple candidate delays may be verified simultaneously, *all* delays for which $Z_\ell \geq \tau_1$ are passed to the second stage. For example, if k delays are verified simultaneously, $K-k$ other delays are evaluated via first stage processing. Verification decisions are based on the statistic

$$\tilde{Z}_\ell = \sum_{i=1}^{J} (\tilde{X}_{g(i)}^\ell)^2 + (\tilde{Y}_{g(i)}^\ell)^2,$$

where $\tilde{X}_i^\ell$ and $\tilde{Y}_i^\ell$ are the in-phase and quadrature outputs (over n_2 chips) for the J correlators involved. If $\tilde{Z}_\ell \geq \tau_2$, the delay is declared correct and acquisition stops. Otherwise, the J correlators begin first-stage processing of a different candidate delay. In the rare case that two or more delays are verified at the same time, the delay with the largest $\tilde{Z}_\ell$ is chosen.

For both approaches, it is possible that all candidate delays are checked without any being declared correct by the verification stage. In this case, the candidate delays are retested in the same order. During retesting, the receiver ensures, as much as possible, that the test statistics for a given candidate delay are obtained from different subbands than those used previously. Note that for $K = 1$ ($J = L$), both the batch verification and the parallel verification schemes are equivalent and actually employ serial acquisition. The two-stage schemes described here can also be easily modified to single-stage schemes by setting $n_2 = 0$ and $\tau_2 = 0$.

4 System Analysis

If the ℓth candidate delay is correct, then given the values of ρ_i and θ_i, X_i^ℓ is Gaussian with mean $n_1 T_C \rho_i \cos\theta_i$ and Y_i^ℓ is Gaussian with mean $n_1 T_C \rho_i \sin\theta_i$. Integrating the joint density function of X_i^ℓ and Y_i^ℓ with respect to the density functions for ρ_i and θ_i yields

$$f_{X_i^\ell Y_i^\ell}(x_i^\ell, y_i^\ell | H_\ell) = \frac{\exp(-((x_i^\ell)^2 + (y_i^\ell))^2/2(n_1^2\beta^2 + \sigma_1^2))}{2\pi(n_1^2\beta^2 + \sigma_1^2)}, \tag{4}$$

where H_ℓ is the hypothesis that the ℓth delay is correct and $\sigma_1^2 = N_0 n_1 T_C/2$ is the noise variance of each correlator output from the first stage.

If the ℓth delay is incorrect, then X_i^ℓ and Y_i^ℓ are Gaussian random variables with mean zero and variance σ_1^2. The conditional joint density function for a noncoherent correlator not matched is then given by

$$f_{X_i^\ell, Y_i^\ell}(x_i^\ell, y_i^\ell | H_j) = \frac{\exp(-((x_i^\ell)^2 + (y_i^\ell)^2)/(2\sigma_1^2))}{2\pi\sigma_1^2}, \; j \neq \ell. \tag{5}$$

The conditional joint density functions of the in-phase and quadrature outputs for the verification stage can be determined similarly. The results are the same as (4) and (5) if σ_1^2 is replaced by $\sigma_2^2 = N_0 n_2 T_C/2$ and n_1 is replaced by n_2.

Note that we have modeled the fading levels in each subband as constant. As a result, successive correlator outputs from a given subband are not independent. The dependencies make an exact analysis of the performance of the acquisition schemes difficult. However, if we model the fading level for successive statistics from a given subband as independent, we can obtain an approximation to the performance of acquisition with batch verification. In the remainder of this section, we present the steps of this analysis.

When using hybrid acquisition with batch verification, a *false alarm* occurs when all the delays under consideration are incorrect, and one is accepted for verification. In the serial scheme, this simplifies to deciding that any incorrect delay is correct. Each of these errors increases the decision time by n_2 chips as a result of the verification correlation time.

Given that the ℓth delay is correct and $j \neq \ell$, the quantity $(X_i^j)^2 + (Y_i^j)^2$ is exponential with mean $2\sigma_1^2$. Furthermore, as a function of i, these quantities are mutually independent. As a result, Z_j is an Erlang random variable with distribution

$$F_{Z_j}(z|H_\ell) = 1 - \sum_{i=0}^{J-1} \frac{\exp(-z/(2\sigma_1^2))z^i}{(2\sigma_1^2)^i i!}, \quad z \geq 0. \tag{6}$$

The probability of false alarm in the first stage is the probability that one or more of the K decision statistics exceeds the threshold τ_1, which is given by $P_F = 1 - [F_{Z_j}(\tau_1|H_\ell)]^K$.

A *miss* occurs when the correct delay is present in the current group of delays being checked, but none are accepted for verification. The $K-1$ decision statistics from incorrect delays each have the distribution function given in (6). Furthermore, given that the ℓth delay is correct, $(X_i^\ell)^2 + (Y_i^\ell)^2$ is an exponential random variable with mean $2(n_1^2\beta^2 + \sigma_1^2)$ for all i, and therefore Z_ℓ has distribution

$$F_{Z_\ell}(z|H_\ell) = 1 - \sum_{i=0}^{J-1} \frac{\exp(-z/(2(n_1^2\beta^2 + \sigma_1^2)))z^i}{(2(n_1^2\beta^2 + \sigma_1^2))^i i!}, \quad z \geq 0. \tag{7}$$

The probability of a miss is equal to the probability that all K decision statistics are less than τ_1 and thus is given by $P_M = F_{Z_\ell}(\tau_1|H_\ell)[F_{Z_j}(\tau_1|H_\ell)]^{K-1}$.

In addition to false alarms and misses, it is possible that 1) the correct delay is present in the current group of delays being tested and 2) one of the *incorrect* delays is accepted for verification. We refer to this event simply as an *error.* Errors are not possible with a purely serial scheme since only one delay is tested at a time. Given that the correct delay is in the current group, the probability of an error is given by $P_E = 1 - P_M - P_C$, where P_C is the probability that the correct delay is chosen. To determine P_C, note that, given that the ℓth delay is correct, the density function for Z_ℓ is

$$f_{Z_\ell}(z|H_\ell) = \frac{z^{J-1}\exp(-z/2(n_1^2\beta^2 + \sigma_1^2))}{(2(n_1^2\beta^2 + \sigma_1^2))^J (J-1)!}, \quad z \geq 0. \tag{8}$$

The desired probability can be found via the integral

$$P_C = \int_{\tau_1}^{\infty} f_{Z_\ell}(z|H_\ell) \left(F_{Z_j}(z|H_\ell)\right)^{K-1} dz.$$

Evaluating the expression gives

$$P_C = 1 - F_{Z_\ell}(\tau_1|H_\ell) - \sum_{i=1}^{K-1} \sum_{k=0}^{i(J-1)} \binom{K-1}{i} (-1)^{i+1} G(k,i),$$

where

$$G(k,i) = \frac{c_{ik}(2(n_1^2\beta^2 + \sigma_1^2))^k(\sigma_1^2)^J}{(J-1)!(i(n_1^2\beta^2 + \sigma_1^2) + \sigma_1^2)^{J+k}} \, \Gamma\left(k + J, \frac{\tau_1(i(n_1^2\beta^2 + \sigma_1^2) + \sigma_1^2)}{2\sigma_1^2(n_1^2\beta^2 + \sigma_1^2)}\right).$$

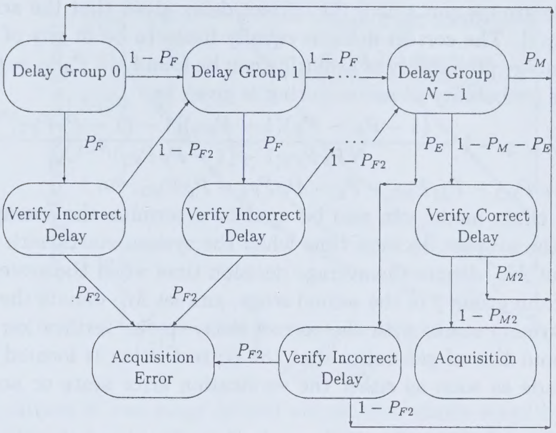


Figure 1: Markov chain model for batch verification hybrid acquisition. The correct delay is assumed to be in group $N-1$.

The coefficient c_{ik} is the kth term of the vector $C(i) = [f_1^{(J)} * f_2^{(J)} * \ldots * f_i^{(J)}]$, where "$*$" represents discrete convolution and $f_j^{(J)}$ represents the vector $[1, \frac{1}{2}, \frac{1}{8}, \ldots, \frac{1}{2^{J-1}(J-1)!}]$. The function $\Gamma(\nu, \mu)$ is the incomplete Gamma function $\Gamma(\nu, \mu) = \int_{\mu}^{\infty} \exp(-t) t^{\nu-1}\, dt$.

If the correct delay is accepted for verification, the second stage may decide that delay to be incorrect. It is straightforward to show that the probability of this event is $P_{M2} = F_{\hat{Z}_\ell}(\tau_2|H_\ell)$, where $F_{\hat{Z}_\ell}(\tau_2|H_\ell)$ is the same as in (7) except that J is replaced by L, σ_1^2 is replaced by σ_2^2, and n_1 is replaced by n_2.

Furthermore, if an incorrect candidate delay is passed to the verification stage, it may be decided that the delay is correct. The probability of verifying an incorrect delay is found to be $P_{F2} = 1 - F_{\hat{Z}_j}(\tau_2|H_\ell)$, where $F_{\hat{Z}_j}(\tau_2|H_\ell)$ is the same as (6) except that J is replaced by L and σ_1^2 is replaced by σ_2^2.

To determine the probability of not acquiring for the hybrid system with batch verification, the Markov chain model of acquisition shown in Fig. 1 is used. We let N denote the number of groups of candidate delays. Recall that there are K delays tested per group, so $N = M/K$, where M is the number of candidate delays. (We assume that M/K is an integer.) For simplicity of exposition, it is assumed without loss of generality that the correct delay is located in the $(N-1)$th delay group. The probability of acquiring given that the acquisition system is in the first stage processing delay group j is $P_{j,A}$. The probability of acquiring given that the verification stage is processing an incorrect delay from group j is $P_{V_j,A}$, and the probability of acquiring given that the correct delay is being processed in the verification stage is $P_{C,A}$.

From the Markov model, the following recursive equations are obtained:

$$\begin{aligned}
P_{j,A} &= (1-P_F)P_{j+1,A} + P_F P_{V_j,A}, \; 0 \le j \le N-2, \\
P_{N-1,A} &= P_M P_{0,A} + P_E P_{V_{N-1},A} + (1 - P_M - P_E) P_{C,A}, \\
P_{V_j,A} &= (1 - P_{F2}) P_{((j+1) \bmod N),A}, \\
P_{C,A} &= 1 - P_{M2}(1 - P_{0,A}).
\end{aligned}$$

These equations can be solved for $P_{j,A}$, $0 \le j \le N-1$, the probability of acquiring for each of the N possible starting positions, or equivalently, the probability of acquiring for each of the N possible groups containing the correct delay given that the acquisition process starts with group 0. The correct delay is equally likely to be in any of the N groups of delays, so the average probability of acquiring can be found via $P_{ACQ} = (1/N)\sum_{j=0}^{N-1} P_{j,A}$. We find that the probability of not acquiring is given by

$$P_{NACQ} = 1 - \frac{(1-P_M-P_E)(1-P_{M2})(1-(1-P_F P_{F2})^N)}{N P_F P_{F2}[1-(1-P_F P_{F2})^{N-1}X]},$$

where $X = P_M + P_{M2} - P_M P_{M2} + P_E - P_E P_{F2} - P_E P_{M2}$.

The Markov chain model can also be used to determine the average decision time. Let M_j denote the average decision time when the system starts with delay group j in the first stage, let M_{V_j} denote the average decision time when the system starts with an incorrect delay from group j in the second stage, and let M_C denote the average decision time when the system starts with the correct delay in the verification stage. Again, it is assumed without loss of generality that the correct delay is located in group $N-1$. A decision is made as soon as either the verification error state or acquisition state is entered.

The set of recursive equations that must be solved to determine the average decision time in chips are

$$\begin{aligned}
M_j &= n_1 + P_F M_{V_j} + (1-P_F)M_{j+1},\ 0 \le j \le N-2,\\
M_{N-1} &= n_1 + P_M M_0 + P_E M_{V_{N-1}} + (1-P_M-P_E)M_C,\\
M_{V_j} &= n_2 + (1-P_{F2})M_{(j+1)\bmod N},\ 0 \le j \le N-2,\\
M_C &= n_2 + P_{M2}M_0.
\end{aligned}$$

Solving for M_ℓ, $0 \le \ell \le N-1$, and using $\mathrm{E}[T_D] = \frac{1}{N}\sum_{j=0}^{N-1} M_j$ gives

$$\begin{aligned}
\mathrm{E}[T_D] &= \frac{n+(1-P_M)n_2}{N(1-A^{N-1}X)} + \frac{[(n+P_F n_2)N + A(n+(1-P_M)n_2)]\,(1-A^{N-1})}{N(1-A)(1-A^{N-1}X)}\\
&\quad + \frac{(n+P_F n_2)(X-1)\left[1-A^{N-1}(N+A-NA)\right]}{N(1-A)^2(1-A^{N-1}X)},
\end{aligned}$$

where $A = 1 - P_F P_{F2}$.

We note that a characteristic function approach can also be used to determine the Z-transform of the decision time density [6]. From this transform, higher moments (including the variance) can be determined. The result for the variance is rather unwieldy and is not included here.

5 Numerical Results and Conclusions

In Fig. 2, the optimal expected decision times of the different two-stage hybrid acquisition schemes are compared to those of a single-stage scheme and the maximum-likelihood decision rule for a 10 carrier system. The receiver has 10 correlators. Each acquisition scheme has been designed to give $P_{NACQ} \le 0.01$, and the parameters n_1, n_2, τ_1, and τ_2 are chosen to minimize the expected acquisition time subject to this constraint. Numerical results are obtained by simulation.

The maximum-likelihood decision rule is given in [7]. It requires that each subband of every candidate delay be correlated for n chips. The correlator outputs are then squared, and the squared outputs corresponding to a given candidate delay are added. The delay is chosen that corresponds to the largest sum.

The figure shows that the two-stage hybrid acquisition methods significantly outperform both the single-stage hybrid scheme and the maximum-likelihood rule. The

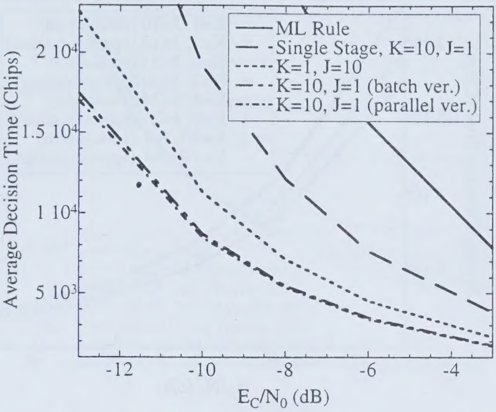


Figure 2: Comparison of two-stage hybrid acquisition, single-stage hybrid acquisition, and maximum-likelihood acquisition, $P_{NACQ} = 0.01$, $M = 100$, $L = 10$.

two-stage schemes perform well because they are able to eliminate many of the delays from consideration quickly. A relatively poor reliability in decision making is permissible in the first stage because the second stage can catch errors. In contrast, the single-stage scheme must perform a lengthy correlation for each candidate delay processed. Maximum-likelihood acquisition gives the worst performance because it must correlate all candidate delays before a decision can be made. (Note that the 100 candidate delays are tested one at a time.)

Fig. 2 also shows, that with regards to the allocation of the 10 correlators, it is better to test 10 different candidate delays simultaneously ($K = 10$) than to test one candidate delay at a time ($K = 1$). In fact, as can be seen in subsequent figures, the performance with $K = 10$ is also better than intermediate values of K (between 1 and 10).

Note that with $K = 10$, the correct delay will likely be missed if it is tested on a subband that is deeply faded. The penalty for missing the correct delay is significant, as it includes the time required to test every other delay. In contrast, with $K = 1$, all ten subbands are used to test a given delay, and it is highly unlikely that all are deeply faded. As the results show, the potential benefits of using $K = 1$ are more than offset by the penalty that results from being able to test fewer candidate delays simultaneously.

The figure also shows that parallel verification is superior to batch verification, although the difference in performance is relatively minor. The results indicate that being able to continuously test candidate delays via first-stage processing is preferable to making quicker decisions in the second stage.

Fig. 3 compares the results of simulation and the approximation derived in Section 4. Here, $P_{NACQ} = 0.01$, $L = 10$, and batch verification is used. It can be seen that the expected times from the approximation are reasonably close to those obtained from simulation. The accuracy of the approximation is somewhat better for large values of K than it is for small values. Note that because the approximation assumes independent fading from test to test, the results also approximate the performance of hybrid acquisition for a channel in which the fading levels vary with time. The closeness between the two sets of results implies that acquisition performance does not change significantly if the fading levels change slowly with time.

Fig. 3 also explicitly shows that performance improves monotonically with increasing

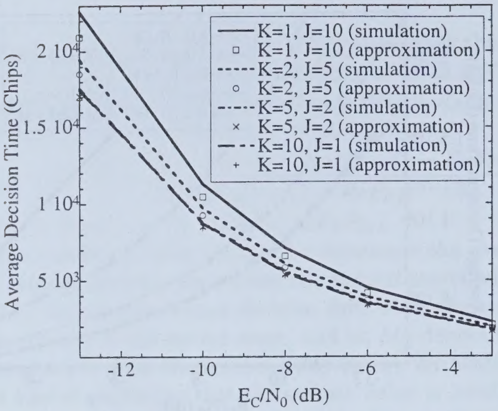


Figure 3: Average decision times for batch verification hybrid acquisition schemes, $P_{NACQ} = 0.01$, $M = 100$, $L = 10$.

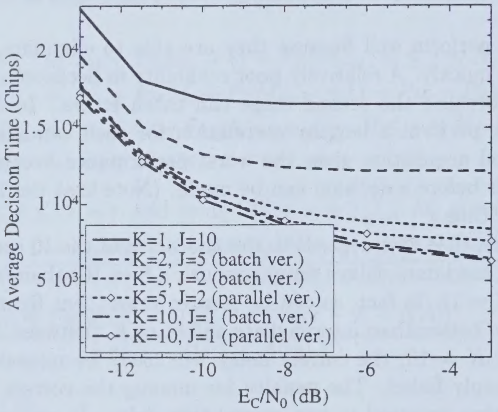


Figure 4: Average decision times for hybrid acquisition schemes designed to give $P_{NACQ} = 0.01$ at $E_C/N_0 = -13$ dB, $M = 100$, $L = 10$. N_0 is fixed.

K. The difference in performance between $K = 5$ and $K = 10$ is negligible, however.

A receiver may be designed to give a minimum level of performance at some worst-case signal-to-noise ratio. It is then of interest to determine how the receiver's performance changes as E_C/N_0 increases. Figs. 4 and 5 show the average decision times and probabilities of not acquiring respectively (as determined by simulation) at various signal-to-noise ratios for different hybrid schemes. The schemes are designed to minimize the expected acquisition time with $P_{NACQ} \leq 0.01$ at $E_C/N_0 = -13$ dB.

The results again show that $K = 10$ is best. Fig. 5 shows that designing for a worst-case value of E_C/N_0 guarantees that $P_{NACQ} \leq 0.01$ even at higher values of E_C/N_0. In addition, a comparison of Figs. 2 and 4 shows that the penalty for using parameters associated with the worst-case E_C/N_0 is not severe. For example, with $E_C/N_0 = -3$ dB (10 dB from the worst-case value) and $K = 10$, the average acquisition time, as compared with the optimized performance, is only worse by a factor of 3.5. The penalty is less if E_C/N_0 is closer to the worst-case value.

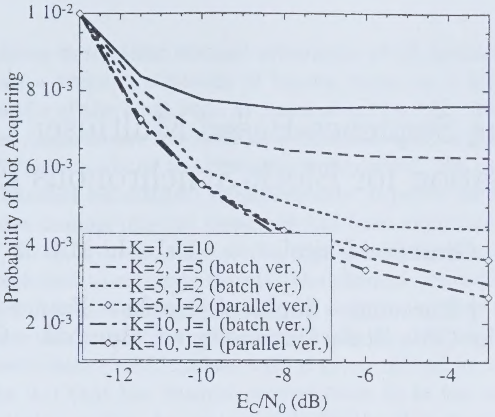


Figure 5: Probability of not acquiring for hybrid acquisition schemes designed to give $P_{NACQ} = 0.01$ at $E_C/N_0 = -13$ dB, $M = 100$, $L = 10$. N_0 is fixed.

In conclusion, several two-stage hybrid acquisition schemes have been presented for multicarrier DS CDMA. It has been shown that parallel verification using only one sub-band per delay in both the first and second stages gives the best performance in terms of acquisition decision time. Hybrid acquisition schemes with batch verification perform slightly worse. The two-stage hybrid acquisition schemes significantly outperform single-stage hybrid acquisition and maximum-likelihood acquisition. Finally, a closed-form approximation of the performance of hybrid acquisition with batch verification has also been presented. The approximation, which assumes independent fading on each test, provides performance close to that of the simulation results.

References

[1] S. Hara and R. Prasad, "Overview of multicarrier CDMA," *IEEE Commun. Mag.*, pp. 126–133, Dec. 1997.

[2] S. Kondo and L. B. Milstein, "Performance of multicarrier DS CDMA systems," *IEEE Trans. Commun.*, vol. COM-44, pp. 238–246, Feb. 1996.

[3] V. V. Veeravalli and C. W. Baum, "Hybrid acquisition schemes for direct sequence CDMA systems," *Intern. J. Wireless Inform. Netw.*, vol. 3, pp. 55–65, Jan. 1996.

[4] M. K. Simon, J. K. Omura, R. A. Scholtz, and B. K. Levitt, *Spread Spectrum Communications Handbook, Revised Edition.* New York: McGraw Hill, 1994.

[5] C. W. Baum and G. S. Hosangadi, "Pipelined sequential acquisition in a DSSS communication system," in *Proc. 1997 IEEE ICPWC*, pp. 189–193, 1997.

[6] A. Polydoros and C. L. Weber, "A unified approach to serial search spread-spectrum code acquisition – Part I," *IEEE Trans. Commun.*, vol. COM-32, pp. 542–549, May 1984.

[7] F. J. Block and C. W. Baum, "Parallel acquisition of multicarrier direct-sequence spread-sprectrum signals," in *Proc. 1999 IEEE Vehic. Tech. Conf.*, 1999.

Training Sequence-Based Multiuser Channel Estimation for Block-Synchronous CDMA

Giuseppe Caire† Urbashi Mitra‡

† Eurecom – Sophie Antipolis – France
‡ The Ohio State University – Columbus – Ohio

Abstract

Uplink channel estimation for a block-synchronous chip-asynchronous CDMA system as proposed for the time-division duplex (TDD) option of 3rd generation cellular systems is considered. Training midambles are employed for joint channel estimation of all users. An *unstructured* approach based on modeling the effective user channels as unknown FIR filters is compared with two *structured* methods that exploit *a priori* knowledge about the user channels such as the maximum delay spread, the transmit chip-shaping pulse and the path delays. A low-complexity, high-performance delay estimator based on maximum-likelihood is proposed which explicitly estimates the delays of the individual multipath components of each user.

Keywords: DS/CDMA, channel estimation, multiuser detection.

1 Introduction

One of the proposals for UMTS, the European 3rd generation mobile communication system, is based on hybrid TDMA/CDMA (briefly denoted as T-CDMA) [1, 2, 3]. In this scheme, multiple access is regulated by a TDMA slot structure similar to GSM [4], and time-division duplexing (TDD) is adopted. User signals are direct-sequence spread-spectrum, with limited processing gain. In standard TDMA, one single user per cell is allowed to transmit over a slot. However, in T-CDMA multiple users per cell are allowed to transmit over the same time slot.

In the uplink (mobile-to-base), users have a coarse common timing reference and are able to align their signal "blocks" with the slot reference of the base station. Due to imperfect synchronization and to the delay-spread of the multipath channels, the signal blocks experience some misalignment. This misalignment is compensated for by inserting guard intervals of appropriate duration. Therefore, the system is block-synchronous, but cannot be considered chip-synchronous.

Provided that the blocks are sufficiently short with respect to the channel coherence time [5], the user channels can be considered time-invariant over each slot. On the other hand, due to the TDMA dynamic user allocation over the slots and to the bursty nature of transmission, tracking the channels from block to block might be infeasible. Hence, we shall consider blockwise channel estimation, where the receiver estimates the user channels block-by-block without tracking across different blocks.

We consider training-based joint channel estimation of all uplink users as in [3]. Each signal block contains a training sequence of known chips, in a fixed nominal position (typically, in the middle of the block [4]). In classical LS (or ML) training-based channel estimation, the user channels are modeled as FIR filters with unknown coefficients [3], and no *a priori* information about the channels is exploited. We refer to this approach as "unstructured" channel estimation. More recently, *a priori* knowledge of the structure of the overall use channel impulse responses has been exploited in order to improve estimation [6, 7, 8, 9, 10, 11]. In some works, the special feature of rectangular chip-shaping pulses is exploited to estimate directly the channel physical parameters (delays and path gains). In this way, a minimal number of unknowns needs to be estimated. On the other hand, these methods do not generalize easily to arbitrary chip-shaping pulses (typically, root-raised-cosine (RRC) pulses with a given roll-off factor α [5]). Other approaches exploit the fact that the channel vectors must lie in the column-space of an *a priori* known convolution matrix determined only by the chip-shaping pulse and by the maximum delay-spread. This approach can be applied to arbitrary approximately bandlimited chip-shaping pulses, but requires in general more unknowns than the methods based on "physical" channel parameterization.

In this paper, we propose two types of "structured" channel estimators which can be applied to any arbitrary (approximately bandlimited) chip-shaping pulse. Our first method is essentially the multiuser version of [10], and exploits only the coarse information represented by the maximum delay-spread and by the chip-shaping pulse. Our second method is based on the "physical" channel parameterization and exploits the knowledge of the path delays for each user. In practice, this information is not available. Thus, we propose a two-step approach where first the path delays are explicitly estimated, and then used in the structured channel estimator. Starting from a ML approach, we derive a low-complexity delay estimator with very good performance, which can be used in the first step of structured channel estimation.

2 Signal model

We consider the uplink of a CDMA system with K block-synchronous users. The baseband receiver front-end is an ideal Low-Pass Filter with (one-sided) bandwidth $W/2$. The k-th user's total channel impulse response $g_k(t)$ is given by the convolution of the chip-shaping pulse $\psi(t)$, common to all users, with the k-th user multipath channel response $c_k(t)$ and the LPF. The pulse $\psi(t)$ is bandlimited with bandwidth $(1+\alpha)/(2T_c) \leq W/2$, such that $\int \psi(\tau)\psi(\tau+t)^* d\tau$ satisfies the Nyquist criterion [5]. The noise $\nu(t)$ after LPF is complex circularly-symmetric Gaussian with autocorrelation function $E[\nu(t)\nu(t-\tau)^*] = N_0 \text{sinc}(W\tau)$.

We do not make an explicit distinction between the delay introduced by non-ideal user synchronization and the delay introduced by multipath propagation. These effects are incorporated in the channel impulse response

$$c_k(t) = \sum_{p=0}^{P-1} c_{k,p}\delta(t-\tau_{k,p}) \tag{1}$$

where we let Δ denote the maximum delay-spread (over all users) accounting for synchronization errors and multipath delay-spread. All users have the same *transmit* average energy per chip: different average *received* signal-to-noise ratios (SNR) are taken into account as an effect of the channels, by multiplying the chanel gains of each user k by the corresponding amplitude factor.

The LPF output $r(t)$ is sampled at rate $W = N_c/T_c$ (N_c is an integer), without any explicit timing reference. We define the polyphase representation of the discrete-time lowpass filtered channel impulse response as $g_{k,\ell}[m] \triangleq g_k((mN_c+\ell)/W)$ for $\ell = 0, \dots, N_c-1$. and we assume that, for all possible channel realizations and user synchronization errors, there exists an integer Q such that $g_{k,\ell}[m]$ is negligible for $m \notin \{0, \dots, Q-1\}$. Under this finite-memory assumption, we define the channel vectors $\mathbf{g}_{k,\ell} = (g_{k,\ell}[0], \dots, g_{k,\ell}[Q-1])^T$.

Users transmit sequences of known chip-symbols of length $M+Q-1$ starting at a given nominal point of the block. Without loss of generality, we let $n = -Q+1$ be the training sequence starting point. In order to estimate the user channels, the receiver forms the vectors of received signal samples

$$\mathbf{r}_\ell = \mathbf{A}\mathbf{g}_\ell + \boldsymbol{\nu}_\ell \tag{2}$$

where $\mathbf{g}_\ell = (\mathbf{g}_{1,\ell}^T, \dots, \mathbf{g}_{K,\ell}^T)^T$ is the total user channel vector corresponding to the sampling phase ℓ; $\boldsymbol{\nu}_\ell = (\nu[\ell], \nu[N_c+\ell], \dots, \nu[(M-1)N_c+\ell])^T$ is a vector of i.i.d. noise samples; and $\mathbf{A} = [\mathbf{A}_1, \dots, \mathbf{A}_K]$ is a $M \times KQ$ block matrix containing only training chip-symbols, whose k-th $M \times Q$ block is given by

$$\mathbf{A}_k = \begin{bmatrix} a_k[0] & a_k[-1] & \cdots & a_k[-Q+1] \\ a_k[1] & a_k[0] & \cdots & a_k[-Q+2] \\ \vdots & & \ddots & \vdots \\ a_k[M-1] & a_k[M-2] & \cdots & a_k[M-Q] \end{bmatrix} \tag{3}$$

3 Unstructured ML channel estimation

Since the noise is i.i.d. and Gaussian, ML estimation is equivalent to the simple LS estimation [12]:

$$\widehat{\mathbf{g}}_\ell = (\mathbf{A}^H\mathbf{A})^{-1}\mathbf{A}^H\mathbf{r}_\ell \tag{4}$$

for $\ell = 0, \dots, N_c-1$, where we assume that $\mathbf{A}$ rank$(\mathbf{A}) = KQ$. We refer to the above estimation method as "unstructured", since we treat $\mathbf{g}_\ell$ as a vector of KQ unknowns.

The resulting normalized estimation MSE of the unstructured estimator is given by

$$\epsilon^2_{\text{unstr}} \triangleq \frac{1}{KQN_c}\sum_{\ell=0}^{N_c-1} E[|\widehat{\mathbf{g}}_\ell - \mathbf{g}_\ell|^2] = \frac{N_0}{KQ}\mathrm{Tr}\left((\mathbf{A}^H\mathbf{A})^{-1}\right) \tag{5}$$

It is well-known [13, 14, 3], that optimal training sequences minimizing $\epsilon^2_{\text{unstr}}$ must satisfy $\mathbf{A}^H\mathbf{A} \propto \mathbf{I}$. The construction of optimal training sequences satisfying $\mathbf{A}^H\mathbf{A} \propto \mathbf{I}$ has been investigated in several papers (see [3, 13, 14, 15, 16, 17] and references therein). Most related research considers sequences constructed from simple symbol alphabets, like BPSK and QPSK. Unfortunately, this requirement is too restrictive and optimal sequences cannot be found for most training lengths M. In this paper, we require the chip-symbols to belong to a N-th root-of-unity alphabet $\mathcal{A}_N = \{e^{j2\pi i/N} : i = 0, \dots, N-1\}$, for some integer N. This choice is enough to obtain optimal training sequences for any desired training length M, while preserving the constant envelope. In particular, we propose to derive the training sequences for all K users from a single Perfect Root-of Unity Sequence (PRUS) [16]:

Definition: PRUS. *The sequence* $\mathbf{x} = (x_0, \dots, x_{M-1}) \in \mathbb{C}^M$ *is a PRUS if* $x_m \in \mathcal{A}_N$, *for all* $m = 0, \dots, M-1$ *and some integer* N, *and if its periodic autocorrelation satisfies*

$$\phi_x(n) = \sum_{m=0}^{M-1} x_m x^*_{[m-n \mod M]} = M\delta_{n,0}$$

$\square$

Let $\mathbf{x}$ be a PRUS of length $M \geq KQ$. Then, the k-th user training sequence $(a_k[-Q+1], \dots, a_k[M-1])$ is obtained from $\mathbf{x}$ as

$$a_k[m] = \sqrt{\mathcal{E}_c} x_{[m-(k-1)Q \mod M]} \tag{6}$$

for all $k = 1, \dots, K$ and $m = -Q+1, \dots, M-1$, where $\mathcal{E}_c$ is the transmit chip-energy. By using (6) into (3), it is easy to check that the columns of $\mathbf{A}$ are distinct cyclic shifts of the same PRUS $\mathbf{x}$. By construction, we obtain $\mathbf{A}^H\mathbf{A} = M\mathcal{E}_c\mathbf{I}$, as desired, and the resulting minimum estimation error is given by

$$\epsilon^2_{\text{unstr}} = \frac{L}{M}\left(\frac{\mathcal{E}_s}{N_0}\right)^{-1} \tag{7}$$

where $\mathcal{E}_s = L\mathcal{E}_c$ is the nominal average transmit energy per symbol and L is the spreading factor (number of chip/symbol).

4 Structured channel estimation

We define the k-th sampled LPF channel response $\widetilde{\mathbf{c}}_k \triangleq (\widetilde{c}_k[0], \dots, \widetilde{c}_k[D-1])^T$ with j-th element $\widetilde{c}_k[j] \triangleq \frac{1}{\sqrt{W}} \int c_k(t)\text{sinc}(j - t/W)dt$, the k-th channel gain vector $\mathbf{c}_k \triangleq (c_{k,0}, \dots, c_{k,P-1})^T$, the $Q \times D$ convolution matrix $\mathbf{\Psi}_\ell$ with (i,j)-th element $\psi\left(\frac{iN_c+\ell-j}{W}\right)$ and the $D \times P$ interpolation matrix $\mathbf{\Phi}_k$ with (j,p)-th element $\text{sinc}\left(j - \tau_{k,p}W\right)/\sqrt{W}$. Then, we let $\widetilde{\mathbf{c}} \triangleq (\widetilde{\mathbf{c}}_1^T, \dots, \widetilde{\mathbf{c}}_K^T)^T$, $\mathbf{c} \triangleq (\mathbf{c}_1^T, \dots, \mathbf{c}_K^T)^T$, $\mathbf{\Phi} \triangleq \text{diag}(\mathbf{\Phi}_1, \dots, \mathbf{\Phi}_K)$ and we define the Kronecker product matrix $\underline{\mathbf{\Psi}}_\ell \triangleq \mathbf{I}_K \otimes \mathbf{\Psi}_\ell$, the block vector $\mathbf{g} \triangleq (\mathbf{g}_0^T, \dots, \mathbf{g}_{N_c-1}^T)^T$ and the block matrix $\underline{\mathbf{\Psi}} \triangleq [\underline{\mathbf{\Psi}}_0^T, \dots, \underline{\mathbf{\Psi}}_{N_c-1}^T]^T$. Eventually, we can write

$$\mathbf{g} = \underline{\mathbf{\Psi}}\widetilde{\mathbf{c}} = \underline{\mathbf{\Theta}}\mathbf{c} \tag{8}$$

Equations (8) define the *a priori* structure of the channel impulse response. The matrix $\underline{\mathbf{\Psi}}$ is determined by the chip-shaping pulse $\psi(t)$ and by the maximum delay-spread Δ, therefore it is always known by the receiver. The matrix $\underline{\mathbf{\Theta}}$ is determined also by the path delays $\{\tau_{k,p} : p = 0, \dots, P-1\}$ for all $k = 1, \dots, K$, and by the number of paths P, which are generally not known by the receiver.

4.1 Type I structured estimation

Let $\mathbf{r} \triangleq (\mathbf{r}_0^T, \dots, \mathbf{r}_{N_c-1}^T)^T$ and $\boldsymbol{\nu} \triangleq (\boldsymbol{\nu}_0^T, \dots, \boldsymbol{\nu}_{N_c-1}^T)^T$. Then,

$$\mathbf{r} = [\mathbf{I}_{N_c} \otimes \mathbf{A}]\,\mathbf{g} + \boldsymbol{\nu} \tag{9}$$

where $\mathbf{A}$ is the same as defined by (3). From the first equality of (8), we have that the desired channel vector $\mathbf{g}$ lies in the column-space of the *a priori* known matrix $\underline{\mathbf{\Psi}}$. Since $KQN_c > KD$, this has a non-trivial null-space and this additional information can be exploited to improve channel estimation. By using the singular-value decomposition (SVD) [18] $\underline{\mathbf{\Psi}} = \mathbf{U}'\mathbf{S}'(\mathbf{V}')^H$, where $\mathbf{S}'$ is $\rho' \times \rho'$ diagonal, ρ' is the rank of $\underline{\mathbf{\Psi}}$, and $\mathbf{U}'$, $\mathbf{V}'$ are rectangular matrices with orthonormal columns and dimension $KQN_c \times \rho'$ and $KD \times \rho'$, respectively, we get the following channel estimator:

1. Obtain the ML estimate of $\mathbf{d}'$ from the observation $\mathbf{r}$ as

$$\widehat{\mathbf{d}} = \arg\min_{\mathbf{d}} \left|\mathbf{r} - [\mathbf{I}_{N_c} \otimes \mathbf{A}]\, \mathbf{U}'\mathbf{d}\right|^2 \tag{10}$$

2. Obtain the Type I structured estimate of $\mathbf{g}$ as $\widehat{\mathbf{g}} = \mathbf{U}'\widehat{\mathbf{d}}$.

Since $\underline{\mathbf{\Psi}}$ is known *a priori*, no real-time SVD computation is required.

4.2 Type II structured estimation

If the delays $\tau_{k,p}$ and the number of paths P are known, the receiver can compute $\underline{\mathbf{\Theta}}$ and impose that the channel vector $\mathbf{g}$ must lie in its column space (see the second equality of (8)). Again, it is convenient to re-parameterize the problem by using the SVD $\underline{\mathbf{\Theta}} = \mathbf{U}''\mathbf{S}''(\mathbf{V}'')^H$, where $\mathbf{S}''$ is $\rho'' \times \rho''$ diagonal, ρ'' is the rank of $\underline{\mathbf{\Theta}}$, and $\mathbf{U}''$, $\mathbf{V}''$ are rectangular matrices with orthonormal columns and dimension $KQN_c \times \rho''$ and $KP \times \rho''$, respectively. Since in general neither the $\tau_{k,p}$'s nor the number of paths P are *a priori* known, these parameters must also be estimated from the received signal. We propose a two-step approach where first an estimate $\widehat{\tau}_{k,p}$ of delays $\tau_{k,p}$ is obtained and then Type II structured estimator is computed assuming $\tau_{k,p} = \widehat{\tau}_{k,p}$. We have:

1. Obtain an estimate the maximum number of paths per user $\widehat{P}$ and of the delays $\{\widehat{\tau}_{k,p} : p = 0, \ldots, \widehat{P} - 1\}$, for all $k = 1, \ldots, K$.

2. Based on the estimated delays, compute an estimate $\widehat{\underline{\mathbf{\Theta}}}$ of $\underline{\mathbf{\Theta}}$ and the $KQN_c \times \widehat{\rho}''$ factor $\widehat{\mathbf{U}}''$ in its SVD.

3. Under the assumption $\mathbf{U}'' = \widehat{\mathbf{U}}''$, obtain the ML estimate of $\mathbf{d}''$ from the observation $\mathbf{r}$ as

$$\widehat{\mathbf{d}} = \arg\min_{\mathbf{d}} \left|\mathbf{r} - [\mathbf{I}_{N_c} \otimes \mathbf{A}]\, \widehat{\mathbf{U}}''\mathbf{d}\right|^2 \tag{11}$$

4. Obtain the Type II structured estimate of $\mathbf{g}$ as $\widehat{\mathbf{g}} = \widehat{\mathbf{U}}''\widehat{\mathbf{d}}$.

Since $\widehat{\underline{\mathbf{\Theta}}}$ is not known *a priori*, a real-time SVD per block is needed. In general, since $\mathbf{U}'' \neq \widehat{\mathbf{U}}''$ the Type II estimator is biased.

Next, we find optimal training sequence sets for Type I and Type II structured estimators (assuming perfect knowledge of the delays for the latter). The solution of the LS estimation (10) and (11) is given by

$$\widehat{\mathbf{d}} = \left(\mathbf{U}^H \left[\mathbf{I}_{N_c} \otimes (\mathbf{A}^H\mathbf{A})\right] \mathbf{U}\right)^{-1} \mathbf{U}^H[\mathbf{I}_{N_c} \otimes \mathbf{A}^H]\mathbf{r}$$

where $\mathbf{U} = \mathbf{U}'$ (resp., $\mathbf{U} = \mathbf{U}''$) for Type I (resp., Type II) estimation. The error vector in the estimation of $\mathbf{d}$ is given by $\mathbf{e} = \widehat{\mathbf{d}} - \mathbf{d}$, $\sim \mathcal{NC}(\mathbf{0}, \boldsymbol{\Sigma})$, where

$$\boldsymbol{\Sigma} \triangleq E[\mathbf{e}\mathbf{e}^H] = N_0 \left(\mathbf{U}^H \left[\mathbf{I}_{N_c} \otimes (\mathbf{A}^H\mathbf{A})\right] \mathbf{U}\right)^{-1}$$

The error vector in the estimation of $\mathbf{g}$ is given by $\mathbf{Ue}$, $\sim \mathcal{NC}(\mathbf{0}, \mathbf{U}\boldsymbol{\Sigma}\mathbf{U}^H)$. The resulting normalized estimation MSE is given by

$$\epsilon^2_{\text{struc}} \triangleq \frac{1}{KQN_c}\text{Tr}\left(\mathbf{U}\boldsymbol{\Sigma}\mathbf{U}^H\right) = \frac{1}{KQN_c}\text{Tr}(\boldsymbol{\Sigma}) \tag{12}$$

where we used the fact that $\mathbf{U}^H\mathbf{U} = \mathbf{I}$. Optimal training sequences should satisfy $\boldsymbol{\Sigma} \propto \mathbf{I}$. Fortunately, since $\mathbf{U}$ has orthonormal columns, we have immediately that if $\mathbf{A}^H\mathbf{A} \propto \mathbf{I}$ then also $\boldsymbol{\Sigma} \propto \mathbf{I}$. We conclude that optimal sequences for unstructured estimation are also optimal for Type I and Type II structured estimation (assuming ideal delay estimation for the latter). In particular, the construction of training sequences from a single PRUS of length $M \geq KQ$, as indicated in (6), can be successfully applied here. The resulting minimum estimation error is given by $\epsilon^2_{\text{struc}} = \frac{\rho}{KQN_c}\epsilon^2_{\text{unstr}}$ where $\rho = \rho'$ (resp., $\rho = \rho''$) for Type I (resp., Type II) estimation. Then, the ratio between the minimum structured and unstructured estimation errors is exactly equal to the ratio between the number of structured and unstructured unknown parameters to be estimated (this does not hold in general, for suboptimal training sequences).

5 Delay estimation

We propose a delay estimator derived from ML estimation of $\boldsymbol{\tau} \triangleq \{\tau_{k,p}\}$ from the observation $\widehat{\mathbf{g}}$, given by the estimated discrete-time low-pass filtered channel impulse responses provided either by unstructured or by Type I structured estimation. This method can be used in the preliminary delay estimation of the Type II structured channel estimator described in Section 4.2. We can write

$$\widehat{\mathbf{g}} = \mathbf{g} + \mathbf{e} = \underline{\boldsymbol{\Theta}}\mathbf{c} + \mathbf{e} \tag{13}$$

where $\mathbf{e} \sim \mathcal{NC}(\mathbf{0}, \boldsymbol{\Sigma})$, independent of the vector of channel gains $\mathbf{c}$. We assume Rayleigh fading, so that $\mathbf{c}$ is $\sim \mathcal{NC}(\mathbf{0}, \Lambda)$. For given delay vector $\boldsymbol{\tau}$, $\underline{\boldsymbol{\Theta}}$ is fixed and $\widehat{\mathbf{g}}$ is conditionally $\sim \mathcal{NC}(\mathbf{0}, \mathbf{R}_g(\boldsymbol{\tau}))$, where the conditional covariance matrix is given by

$$\mathbf{R}_g(\boldsymbol{\tau}) = \underline{\boldsymbol{\Theta}}(\boldsymbol{\tau})\Lambda\underline{\boldsymbol{\Theta}}(\boldsymbol{\tau})^H + \boldsymbol{\Sigma}$$

where we indicate explicitly the dependence on the delay vector $\boldsymbol{\tau}$. With the above statistical model, the ML estimate of $\boldsymbol{\tau}$ based on the observation $\widehat{\mathbf{g}}$ is given by

$$\widehat{\boldsymbol{\tau}} = \arg \max_{\boldsymbol{\tau}} \left\{-\widehat{\mathbf{g}}^H\mathbf{R}_g^{-1}(\boldsymbol{\tau})\widehat{\mathbf{g}} - \log \text{Det}(\mathbf{R}_g(\boldsymbol{\tau}))\right\} \tag{14}$$

This requires the maximization of the log-likelihood function over a KP-dimensional real space. Moreover, the log-likelihood function depends on $\boldsymbol{\tau}$ via a matrix inversion.

In order to decrease complexity, we make some simplifications and assumptions. In particular, we assume that: i) white error vector (i.e., $\boldsymbol{\Sigma} = \sigma_e^2\mathbf{I}$); ii) channel gains for different users and for different delays of the same user are mutually independent (i.e., Λ is diagonal); iii) In order to obtain an additional simplification, we assume that the delays τ_k are sufficiently separated, so that $\psi(t-\tau_{k,p})$ and $\psi(t-\tau_{k,q})$ have essentially disjoint support for $p \neq q$. Then, it is possible to show [19] that the ML delay estimator for each user k can be put in the form:

$$\widehat{\tau}_k = \arg \max_{\boldsymbol{\tau}_k} \sum_{p=0}^{P-1} \frac{\sigma_{k,p}^2}{\sigma_e^2 + \sigma_{k,p}^2} \left|\boldsymbol{\psi}_{k,p}^H \widehat{\mathbf{g}}^{(k)}\right|^2 \tag{15}$$

where $\boldsymbol{\tau}_k$ and $\mathbf{g}^{(k)}$ are the subvectors of $\boldsymbol{\tau}$ and of $\mathbf{g}$ corresponding to user k, where $\sigma_{k,p}^2 \triangleq E[|c_{k,p}|^2]$ and where we define the column vector $\boldsymbol{\psi}_{k,p}$ with i-th element

$$[\boldsymbol{\psi}_{k,p}]_i = \psi(i/W - \tau_{k,p})/\sqrt{W} \tag{16}$$

for $i = 0, \dots, QN_c - 1$. We have not escaped a P dimensional maximization and typically independent maximization of each term in (15) yields $\widehat{\tau}_{k,p} = \widehat{\tau}$, for all $p = 0, \dots, P-1$, where $\widehat{\tau}$ is located with high probability in the vicinity of the maximum peak of the sampled observed channel response $\widehat{\mathbf{g}}^{(k)}$. Also, we observe that in practice, both the delay-intensity profile and P are unknown. However, we can still propose a further approximated algorithm which requires only a sequence of P one-dimensional maximizations.

Assume P known. First, define a delay discretization step $\Delta\tau$, such that $T_c/\Delta\tau = N_\tau$ is an integer multiple of N_c. Then, for all $j = 0, \dots, QN_\tau - 1$, define the vectors $\mathbf{v}_j$ of length QN_c with i-th component

$$[\mathbf{v}_j]_i = \psi(i/W - j\Delta\tau)/\sqrt{W}$$

for $i = 0, \dots, QN_c - 1$. Clearly, $\mathbf{v}_j = \boldsymbol{\psi}_{k,p}$ if $j\Delta\tau = \tau_{k,p}$, for some j, while if $\tau_{k,p}$ is not an integer multiple of $\Delta\tau$, the maximum delay discretization error is $\Delta\tau/2$. Initialize the vector $\mathbf{w}_0 = \widehat{\mathbf{g}}^{(k)}$ and the set of delay indexes $\mathcal{S}_0 = \{0, \dots, QN_\tau - 1\}$. For $p = 0, \dots, P-1$, repeat the following steps:

1. Estimate the p-th delay as $\widehat{\tau}_{k,p} = \widehat{j}_p \Delta\tau$, where

$$\widehat{j}_p = \arg \max_{j \in \mathcal{S}_p} \left|\mathbf{v}_j^H \mathbf{w}_p\right|^2 \tag{17}$$

2. Eliminate the effect of the p-th delay from the observed channel impulse response as

$$\mathbf{w}_{p+1} = \mathbf{w}_p - \left(\frac{\mathbf{v}_{\widehat{j}_p}^H \mathbf{w}_p}{|\mathbf{v}_{\widehat{j}_p}|^2}\right) \mathbf{v}_{\widehat{j}_p} \tag{18}$$

3. Update the delay index set as

$$\mathcal{S}_{p+1} = \mathcal{S}_p - \{\widehat{j}_p - N_\tau + 1, \dots, \widehat{j}_p + N_\tau - 1\} \tag{19}$$

6 Performance with linear detectors

In order to investigate the impact of channel estimation on the receiver performance, we consider some simple FIR (or "one-shot") linear receivers. After low-pass filtering and sampling at rate W, for each n, a window of samples centered around the interval $[nLT_c, (n+1)LT_c]$ is used to detect the n-th symbol. Since transmission is chip-asynchronous and the channel delay-spread can be larger than a symbol interval, the processing window should span more than one symbol interval. Approximated single-user matched filters (SUMF) and linear minimum mean-square error (LMMSE) filters [20] are obtained from the channel estimates produced by the algorithms seen before.

We consider a system with $K = 8$ users, spreading factor $L = 16$, RRC chip-shaping pulse with roll-off $\alpha = 0.22$, truncated over $\kappa = 12$ chips, and maximum channel delay spread of $\Delta = 20T_c$. With this delay-spread, user blocks may be misaligned by more than one symbol and ISI is not at all negligible, in contrast to what is normally assumed in most DS/CDMA literature. The maximum channel length is $Q = 20 + 12 = 32$ chips. We choose training length $M = 256$, i.e., the minimum length necessary for estimating 8 channels of length 32. As a performance measure, we consider the signal-to-interference plus noise ratio (SINR) at the filter output. Since the channel impulse responses change randomly and independently from block to block, the output SINR is a random variable. Therefore, we use Monte Carlo simulation to evaluate the empirical SINR cumulative distribution function (cdf).

Figs. 1 show the SINR cdf for the SUMF receiver in the case of slow and fast power control. In the case of slow power control, the ratio E_k/N_0 is set to 10 dB for all users and no further normalization of the path gains applied. Then, the average received SNR on the shortest path (for which $\sigma_{k,0}^2 = 1$) is actually equal to 10 dB for all users, but the instantaneous user received power fluctuates considerably from block to block, because of the uncompensated Rayleigh fading. In the case of fast power control, E_k/N_0 is set as before but the channel gains are normalized such that $\sum_{p=0}^{P-1} |c_{k,p}|^2 = 1$. This could be obtained in practice by a TDD system where each user measures the signal power received on the downlink and exploits reciprocity in order to compensate instantaneously for the Rayleigh fading block by block. The SINR cdf yields the block outage probability, defined as the probability that the SINR is below a fixed threshold γ. The horizontal dashed line corresponds to $F_{\text{sinr}}(\gamma) = 10^{-1}$.

Figs. 2 show analogous results for the LMMSE receiver. Apparently, structured channel estimaiton does not provide considerable advantages with SUMF detection. On the contrary, the MMSE detector is more sensitive to accurate channel estimation and structured methods provide considerable improvements in terms of outage probability.

References

[1] ETSI SMG2#24, "WB-TDMA/CDMA system description performance evaluation," Madrid, Spain, Dec. 15 - 19, 1997.

[2] A. Klein, G. Kawas Kaleh, P. Baier, "Zero forcing and minimum mean-square error equalization for multiuser detection in code-division multiple-access channels," *IEEE Trans. on Vehic. Tech.*, Vol. 45, No. 2, pp. 276 - 287, May 1996.

[3] B. Steiner, P. Jung, "Optimum and suboptimum channel estimation for the uplink of CDMA mobile radio systems with joint detection," *European Trans. on Commun.*, Vol. 5, No. 1, pp. 39 - 49, Jan.–Feb., 1994.

[4] T. Rappaport, *Wireless Communications.* Englewood Cliffs, NJ: Prentice-Hall, 1996.

[5] J. G. Proakis, *Digital Communications.* New York, NY: McGraw Hill, 2nd ed., 1989.

[6] S. Bensley, B. Aazhang, "Subspace-based channel estimation for Code Division Multiple Access communication systems," *IEEE Trans. on Commun.*, Vol. 44, No. 8, pp. 1009 - 1020, Aug. 1998.

[7] S. Bensley, B. Aazhang, "Maximum-Likelihood synchronization of a single user for Code Division Multiple Access communication systems'," *IEEE Trans. on Commun.*, Vol. 46, No. 3, pp. 392 – 399, March 1996.

[8] E. Strom, S. Parkvall, S. Miller, B. Ottersten, "Propagation delay estimation in asynchronous Direct-Sequence Code-Division Multiple-Access systems," *IEEE Trans. on Commun.*, Vol. 44, No. 1, pp. 84 – 93, Jan. 1996.

[9] E. Ertin, U. Mitra, S. Siwamogsatham, "Maximum-Likelihood based multipath channel estimation for code-division multiple-access systems," submitted to the *IEEE Trans. on Commun.*, June 1998.

[10] B. C. Ng, M. Cedervall, A. Paulraj, "A structured channel estimation for maximum likelihood sequence detection," *IEEE Commun. Letters,* pp. 52 – 55, March 1997.

[11] B. C. Ng, D. Gesbert, A. Paulraj, "A semi-blind approach to structured channel equalization," Proc. of *IEEE Intern. Conf. on Commun. (ICC) 1997,* pp. 3385 – 3388, Atlanta, 7 – 11 June, 1998.

[12] S. Kay, *Fundamentals of statistical signal processing: Estimation theory.* New York: Prentice-Hall, 1993.

[13] S. Crozier, D. Falconer, S. Mahmoud, "Least sum of squared errors (LSSE) channel estimation," *IEE Proc.- F,* vol. 138, pp. 371 – 378, Aug. 1991.

[14] A. Clark, Z. Zhu, J. Joshi, "Fast start-up channel estimation," *IEE Proc.- F,* vol. 131, pp. 375 – 382, July 1984.

[15] S. Grant, J. Cavers, "Multiuser channel estimation for detection of cochannel signals," Proceedings of *1999 IEEE Intern. Conf. on Commun. ICC '99,* Vancouver, BC, Canada, 6 – 10 June, 1999.

[16] W. Mow, *Sequence design for spread spectrum,* The Chinese University Press, Chinese University of Hong Kong, Shatin, Hong Kong, 1995.

[17] J. Ng, K. Ben Letaief, R. Murch, "Complex optimal sequences with constant magnitude for fast channel estimation initialization," *IEEE Trans. on Commun.*, Vol. 46, No. 3, pp. 305 – 308, March 1998.

[18] G. Golub, C. Van Loan, *Matrix computations.* The John Hopkins University Press, 2nd ed., 1989.

[19] G. Caire, U. Mitra, "Structured multiuser channel estimation for block-synchronous DS/CDMA," submitted to *IEEE Trans. on Commun.*, July 1999.

[20] S. Verdú, *Multiuser Detection.* New York, NY: Cambridge University Press, 1998.

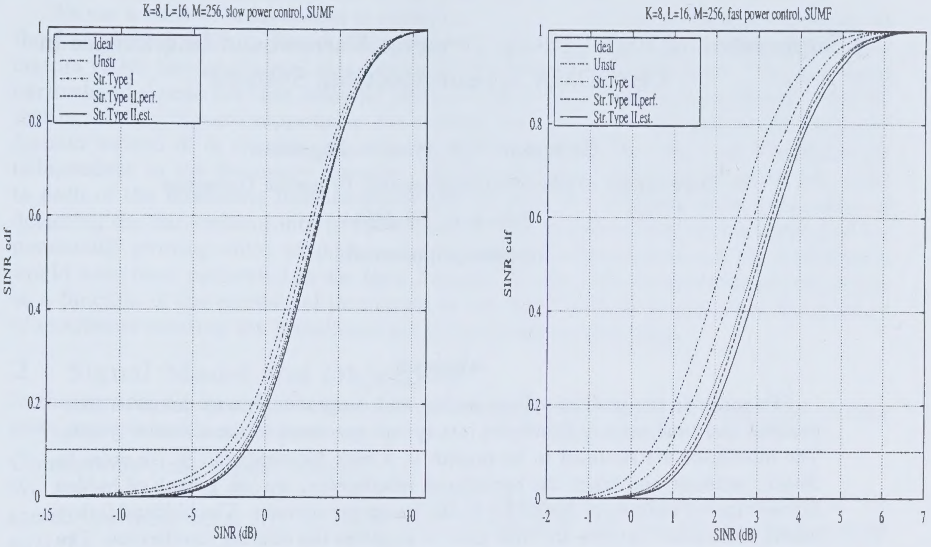


Figure 1: SINR cdf with slow (left) and fast (right) power control and SUMF receiver.

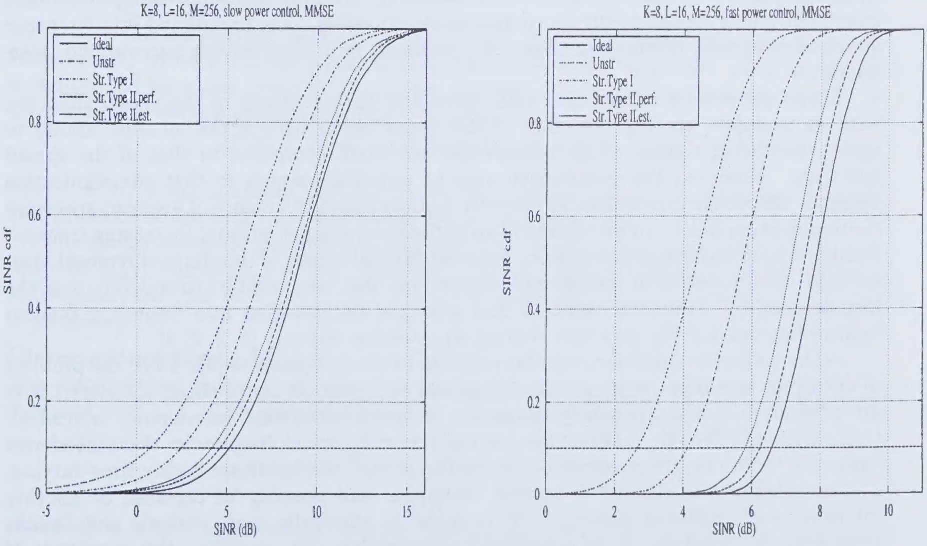


Figure 2: SINR cdf with slow (left) and fast (right) power control and LMMSE receiver.

Suppression of High-Density, Dynamic Narrowband Interference in DS/CDMA Spread-Spectrum Systems*

C. Carlemalm, H.V. Poor, A. Logothetis
Department of Electrical Engineering, Princeton University
Princeton, NJ 08544
cath@ee.princeton.edu

Abstract

We consider the problem of estimating and suppressing many unknown independent and time-varying interferers in a spread-spectrum communication system. The interferers are assumed to be present in a wide frequency range. In order to detect, estimate and track the broadband interference, we use a bank of hidden Markov model estimators operating in the frequency domain. The hidden Markov model estimators' outputs are then used to suppress the existing interference. The computational complexity of our scheme is only linear in the number of interferers. The simulation studies show that our proposed novel schemes adapt quickly in tracking the time-varying nature of the broadband interference.

1 Introduction

In this paper, we consider the problem of detecting, tracking and suppressing broadband interference in a spread-spectrum communication system. The broadband interference is assumed to consist of many unknown, independent and time-varying narrowband interferers.

There are several advantages with spreading the spectrum of the signal that you wish to transmit; see *e.g.* [1]. One of the major advantages is the inherent ability to reject interfering signals whose bandwidths are small compared to that of the spread spectrum. However, the interference may be powerful enough so that communication becomes effectively impossible. In order to improve the performance of a spread-spectrum communication link in an interference corrupted environment without increasing transmit bandwidth (or reducing throughput), some additional means of interference removal must be used. The problem of interference suppression has been studied extensively over the last decades [1]. Previous work in this area can be classified into frequency domain approaches, see *e.g.* [2], and time domain approaches, see *e.g.* [1, 3, 2, 4].

In this paper, we consider a similar problem to the one posed in [4]. Thus, the problem of detection and suppression of *many* unknown independent narrowband interferers that are present in a wide frequency range, *i.e.* forming broadband interference is studied. Contrary to [4], though, it is here assumed unknown for which frequencies these interferers exist. Furthermore, the presence and statistics of the interferers are highly time-varying, *i.e.*, an interferer is allowed to appear, disappear and possibly be replaced by another interferer with different statistics. This model is physically more realistic and flexible than a model assuming fixed broadband interference. By modeling the amplitude of

---

*This research was supported in part by the U.S. Office of Naval Research under Grant N00014-94-1-0115, and in part by the New Jersey Center for Mobile Telecommunications.

each narrowband interferer with an independent Markov chain as indicator functions, the possible presence of the interferers is described efficiently.

We use a bank of independent hidden Markov model (HMM) estimators operating in the frequency domain to detect and estimate the interference. Each of the HMM estimators yields the conditional mean estimate of a narrowband interferer. The estimated narrowband signals are then removed prior to transforming back to time domain. By detecting, estimating and suppressing the existing broadband interference in the frequency domain instead of in the time domain, we make use of the fact that the interferers are independent in the frequency domain. We can therefore apply an adaptive estimator to each of the interferers independently and thereby, the computational complexity of detecting the narrowband interferers is linear in the number of interferers instead of exponentially growing which would have been the case if the estimation of the interference would have been performed in the time domain. Keeping the computational complexity as a function of the number of interferers as low as possible is crucial since the number of interferers creating the broadband interference can be very high.

2 Signal Model and Objectives

In this section, we describe the model for the received noise corrupted signal and we then state our estimation objectives.

Continuous-time Signal Model

We assume that the continuous-time received signal, $y(t)$, consists of the sum of the spread spectrum signal $s(t)$, the broadband interference $x(t)$, and the observation noise $w(t)$, *i.e.*,

$$y(t) = s(t) + x(t) + w(t). \tag{1}$$

The interference $x(t)$, is assumed to consist of at the most M superimposed independent narrowband interferers, $x^m(t)$, $m = 1, 2, \dots M$, where M is a known integer, *i.e.*

$$x(t) = \sum_{m=1}^{M} x^m(t). \tag{2}$$

[5] models the interference as a sum of sinusoids. Here, we extend this model by considering $x(t)$ to be a superposition of amplitude modulated sinusoids. Each amplitude is modulated by a two state Markov chain serving as an indicator function in order to model the interferer's possible presence. In particular,

$$x^m(t) = A^m(t) \exp\{j2\pi mFt\},\ m = 1, 2, \dots M, \tag{3}$$

where $A^m(t)$ is an unknown stochastic random variable indicating the signal strength and presence of the mth interferer at the known frequency mF.

The spread spectrum signal $s(t)$ is assumed to be transmitted over the entire frequency range $[0, f_{max}]$, where $f_{max} = MF$, while the mth narrowband interferer, $x^m(t)$, operates at frequency mF, for $m = 1, 2, \dots M$. Thus, the interferers are assumed to be equispaced in frequency.

Discrete-time Signal Model

The discrete-time model is the result of sampling the continuous-time model in (1) at sampling rate $1/T_s$. The discrete-time observations are given as follows

$$y_k = s_k + \sum_{m=1}^{M} A_k^m \exp\{j2\pi mFkT_s\} + w_k, \tag{4}$$

where w_k is a zero mean white circular Gaussian process with variance σ_w^2. The sampling period, T_s, is assumed to be equal to the chip period, T_c, of the spread spectrum signal. The condition $T_s = T_c$ ensures the sampled spread spectrum signal s_k to be an independent identically distributed (iid) process. A_k^m, $m = 1, 2, \dots M$ is a discrete-time, homogeneous, first-order, 2-state Markov chain with states $q^m = (q_1^m, q_2^m)$, where $q_1^m = 0$ and q_2^m is an unknown non-zero complex number, *i.e.* A^m serves as an indicator function for the presence of the mth interferer. Thus, when the mth Markov chain A_k^m is at state q_1^m, it implies that there is no interferer present at frequency mF. On the other hand, when A_k^m is equal to q_2^m, then the interfering amplitude is q_2^m. q_2^m is allowed to be time-varying. The transition probability matrix of A^m is denoted by $\Pi^m = (\pi_{np}^m)$, where π_{np}^m denotes the probability that A_{k+1}^m is in state q_p^m given that A_k^m was in state q_n^m. That is,

$$\pi_{np}^m = P(A_{k+1}^m = q_p^m | A_k^m = q_n^m),\ n, p \in \{1, 2\}. \qquad (5)$$

Of course, $\pi_{np}^m \geq 0$ and $\sum_{p=1}^{2} \pi_{np}^m = 1$, for each $n, p \in \{1, 2\}$ and $m \in \{1, 2, \ldots M\}$. Let $\bar{\pi}^m$, $m = 1, \ldots M$ denote the initial state probability vector that is the vector containing the probabilities of the states of A_1^m. That is,

$$\bar{\pi}^m = (\bar{\pi}_n^m),\ \forall m = 1, 2, \ldots M, \text{ where } \bar{\pi}_n^m = P(A_1^m = q_n^m)\ \forall m = 1, 2, \ldots M. \qquad (6)$$

Objectives

Given the discrete-time observations y, the objective is to propose both off-line and on-line algorithms that

- are of low complexity. The computational complexity should increase linearly with the number of possible interferers.
- detect accurately which frequencies mF for $m = 1, 2, \ldots M$ are corrupted with interference. Using the model description in Section 2, this is equivalent to estimating the state of the Markov chain A^m for $m = 1, 2, \ldots M$.
- estimate accurately with what probability the mth interferer x^m, $m = 1, 2, \ldots M$, is present.
- estimate accurately the interferers x^m, $\forall m = 1, 2, \ldots M$. Using the model description in Section 2, this is equivalent to determining q_2^m, $\forall m = 1, 2, \ldots M$.
- estimate accurately the transition probabilities from absence to presence of the interferers, *i.e* determine Π^m, $\forall m = 1, 2, \ldots M$ in our model description above.
- suppress the broadband interference, x.

3 Algorithms for Broadband Interference Estimation and Suppression

In this section, we propose off-line and on-line schemes for detection, tracking, estimation and suppression of broadband interference in a spread-spectrum communication system.

The key idea of the suggested algorithms is to perform adaptive filtering in the frequency domain using a bank of hidden Markov model (HMM) estimators (filters for the on-line scheme and smoothers for the off-line scheme) to detect and estimate the interference. The outputs of the HMM filters/smoothers are then used to determine the parameters of our proposed suppression scheme which is also run in the frequency domain. We choose to perform the adaptive estimation in the frequency domain and not in the time domain in order to make use of the fact that the interferers are independent in the frequency domain. Therefore, the computational complexity of detecting the narrowband interferers is linear in the number of interferers instead of exponentially growing which would have been the case if the estimation of the interference would have been performed in the time domain. Keeping the computational complexity as a function of the number of interferers as low as possible is crucial since the number of interferers creating the broadband interference can be very high.

In Fig. 1, our proposed scheme is depicted. The output data are grouped into batches of T consecutive blocks. Each block contains N data points. The discrete Fourier transform (DFT) of the bth batch is given by

$$Y(kF_s, b) = T_s \sum_{n=bN-N}^{bN-1} y_n \exp\left\{-j\frac{2\pi n}{N}k\right\},\ k = 0, \ldots N-1,\ b = 1, \ldots T, \qquad (7)$$

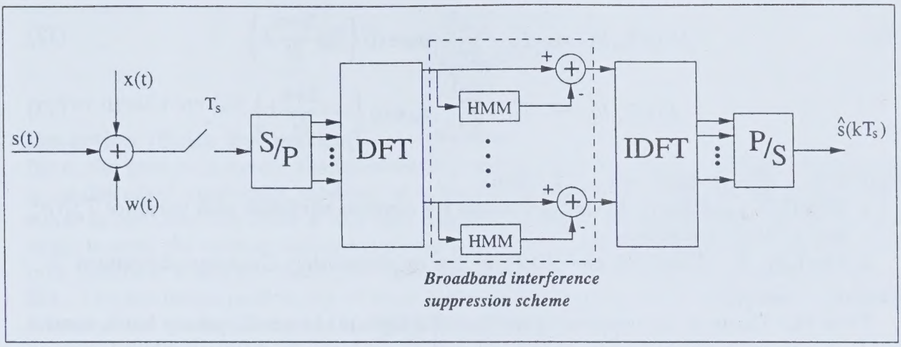


Figure 1: Schematics of the proposed broadband interference suppression schemes using a bank of HMM estimators.

where $F_s = 1/(NT_s)$. The received signal is, as stated in Section 2, a sum of the transmitted spread spectrum signal, s, the observation noise, w, and the interference, x. The DFTs of each of these components are as follows.

$$\begin{aligned} X(kF_s, b) &= T_s \sum_{n=bN-N}^{bN-1} \sum_{m=1}^{M} A_n^m \exp\{j2\pi mFnT_s\} \exp\left\{-j\frac{2\pi n}{N}k\right\}, \\ & \qquad k = 0, \ldots N-1,\ b = 1, \ldots T. \end{aligned} \tag{8}$$

A^m is assumed to be constant in the bth data batch. It is denoted as A_b^m. Furthermore, the interferers x^m, $m = 1, 2, \ldots M$ are, as previously stated, assumed to be mutually independent in the frequency domain. Assuming the interference is constant during one batch we have

$$X^m(kF_s, b) = T_s A_b^m \sum_{n=bN-N}^{bN-1} \sum_{m=1}^{M} \exp\left\{j\frac{2\pi n}{N}\left(\frac{mF}{F_s} - k\right)\right\}, k = 0, \ldots N-1,\ b = 1, \ldots T. \tag{9}$$

By choosing the sampling period T_s and the number of data points N per batch, such that $F_s = F$ and $N = M$, we have the following result

$$X^m(kF_s, b) = \begin{cases} NT_s A_b^m & \text{if } m = k \\ 0 & \text{otherwise} \end{cases} \quad k = 0, \ldots N-1,\ b = 1, \ldots T.$$

The DFT of the observation noise is given by

$$W(kF_s, b) = T_s \sum_{n=bN-N}^{bN-1} w_n \exp\left\{-j\frac{2\pi n}{N}k\right\},\ k = 0, \ldots N-1,\ b = 1, \ldots T. \tag{10}$$

and finally, the DFT of the spread spectrum signal is

$$S(kF_s, b) = T_s \sum_{n=bN-N}^{bN-1} s_n \exp\left\{-j\frac{2\pi n}{N}k\right\},\ k = 0, \ldots N-1,\ b = 1, \ldots T. \tag{11}$$

Theorem 3.1 *Let w_k be a sequence of independent and identically distributed (iid), zero mean white circular Gaussian random variables with finite variance σ_w^2. Let s_k be a sequence of iid random variables with zero mean and finite variance σ_s^2. If the sequences $W(kF_s, b)$ and $S(k, F_s, b)$ are computed as follows*

$$W(kF_s, b) = T_s \sum_{n=bN-N}^{bN-1} w_n \exp\left\{-j\frac{2\pi n}{N}k\right\} \quad (12)$$

$$S(kF_s, b) = T_s \sum_{n=bN-N}^{bN-1} s_n \exp\left\{-j\frac{2\pi n}{N}k\right\} \quad (13)$$

for $k = 0, \dots N-1$ *and* $b = 1, \dots T$, *then*

- $W(kF_s, b)$ *and* $S(kF_s, b)$ *are zero mean iid random variables with variance* $T_s^2 N\sigma_w^2$ *and* $T_s^2 N\sigma_s^2$, *respectively.*
- *For large* N, $W(kF_s, b)$ *and* $S(kF_s, b)$ *are approximately Gaussian distributed.*

PROOF. See [12]. □

From the Theorem above, we observe that the signals at each frequency bin k, consist of 1) a finite-state Markov chain $X(kF_s, b)$ and 2) for large N, additive white Gaussian noise. Hidden Markov theory can thus be used to extract the Markov chains impeded in the additive background noise. In the following two subsections, we use off-line and on-line Maximum likelihood techniques for estimating and subtracting the interference.

Notation: For notational convenience, we choose to denote $Y(mF_s, b)$ at the frequency mF as Y_b^m. We denote the set $\{Y_b^m,\ b = 1, \dots T\}$ as $\mathcal{Y}^m$ and $\{Y_b^m,\ b = b_1, \dots b_2\}$ as $\mathcal{Y}_{b_1:b_2}^m$. Similar notation holds for $S(mF_s, b), X(mF_s, b)$ and $W(mF_s, b)$. Furthermore, we let $\mathcal{A}^m$ denote $\{A_b^m,\ b = 1, \dots T\}$. Let $\theta^m \triangleq \{q_2^m, \Pi^m\}$, $m = 1, \dots M$.

3.1 Off-line Broadband Interference Suppression Scheme

We use a bank of M *independent* HMM smoothers in the frequency domain, that estimate, detect and subtract the interference present at each frequency bin. The estimation and detection of the mth interferer is presented here.

Interference Parameter Estimation

The EM algorithm proposed in [11] is used to obtain the maximum likelihood (ML) estimate of θ^m which is denoted here as $(\theta^m)^{ML}$. As a by-product of the E-step, conditional mean estimates of the state A^m of the mth interference is obtained. It is shown in [6] that under mild regularity conditions, the sequence $\left\{\theta^{m,l}\right\}$ for $l = 1, 2, \dots$, of the EM algorithm converges to a stationary value of the likelihood function. The EM algorithm is as follows.

0. Determine the initial estimate $\theta^{m,1}$.

1. Expectation step: $Q^m\left(\theta^m, \theta^{m,l}\right) \triangleq \mathrm{E}\left\{\ln f(\mathcal{Y}^m, \mathcal{A}^m \mid \theta^m) | \mathcal{Y}^m, \theta^{m,l}\right\}$.

2. Maximization step: $\theta^{m,l+1} = \arg\max_{\theta^m} Q^m\left(\theta^m, \theta^{m,l}\right)$.

3. $l := l + 1$. Iterate steps 1–3 until $||\theta^{m,l+1} - \theta^{m,l}|| < \varepsilon$, where ε is some specified constant.

The expectation step yields

$$\gamma_b^{m,l+1}(i) \triangleq P(A_b^m = i | \mathcal{Y}^m, \theta^{m,l}) \quad (14)$$

and

$$\gamma_b^{m,l+1}(i,j) \triangleq P(A_b^m = j, A_{b-1}^m = i | \mathcal{Y}^m, \theta^{m,l}) \quad (15)$$

for $b = 1, 2, \dots T$, $i, j \in \{1, 2\}$.

The maximization step yields the Markov chain parameters

$$\pi_{ij}^{m,l+1} = \frac{\sum_{b=1}^T \gamma_{b-1}^{m,l}(i,j)}{\sum_{b=1}^T \gamma_b^{m,l}(i)}, \qquad i, j \in \{1, 2\}. \quad (16)$$

and the coefficient, q_2^m,

$$q_2^{m,l+1} = \frac{\sum_{b=1}^{T} Y_b^m \gamma_b^{m,l}(2)}{\sum_{b=1}^{T} T_s N \gamma_b^{m,l}(2)}. \tag{17}$$

For details see [12].

Detection (State Estimation)
Here, the goal is to detect the interferer's possible presence. Each interferer's amplitude is, as discussed previously, modeled as a 2-state Markov chain with one state-level being equal to zero and the other a non-zero complex number. When the state of the Markov chain is zero, the corresponding interferer is absent. Thus, by estimating the state of the mth interferer's Markov chain, we can conclude whether the mth interferer is present or not. The minimum probability of error estimate of detection of the interferer is given as a by-product of the E-step of the HMM smoother as follows

$$(\hat{A}_b^m)^{MAP} = \arg\max_{A_b^m} f\left(A_b^m \mid \mathcal{Y}^m, \theta^{m,l}\right) = \arg\max_{j} \gamma_b^{m,l}(j) \tag{18}$$

Interference Suppression
The outputs of the HMM smoother, $\gamma^{m,l}, q_2^{m,l}$, are used to estimate the mth interferer. The estimate of the narrowband interference at frequency mF_s on the bth data batch is given by the conditional mean estimate

$$\hat{X}_b^{m,l} \triangleq E\{A_b^m | \mathcal{Y}^m, \theta^{m,l}\}, \tag{19}$$

where $\hat{X}_b^{m,l}$ denotes the estimate of X_b^m on the lth iteration. (19) reduces to

$$\hat{X}_b^{m,l} = \gamma_b^{m,l}(2) q_2^{m,l}, \; b = 1, \dots T. \tag{20}$$

Equation (20) represents the estimate of the narrowband interferer at the mth frequency bin, which is computed by weighting the estimate of the interferer's signal strength by the probability that the interferer is active. The broadband interference suppression is performed by subtracting the estimated interferer $\hat{X}_b^m$ from the data Y_b^m of the mth frequency bin, prior to taking the inverse discrete Fourier transform (IDFT). The spread spectrum signal after interference suppression is given by

$$\hat{s}_k = \frac{1}{NT_s} \sum_{m=0}^{N-1} \left(Y_{\mathrm{int}(k/N)+1}^m - \gamma_b^{m,l}(2) q_2^{m,l} \right) \exp\left\{ j \frac{2\pi}{N} mk \right\}, \; k = 0, 1, \dots TN - 1, \tag{21}$$

where $\mathrm{int}(k/N)$ denotes the integer part of the division k/N.

3.2 On-line Broadband Interference Suppression Scheme

We use a bank of M *independent* HMM filters in the frequency domain, that estimate, detect and subtract the interference present at each frequency bin. The HMM filters yield the un-normalized aposteriori probabilities

$$\alpha_b^{m,l}(j) \triangleq f^m(Y_b^m, A_b^m = j | \theta^{m,l})$$

The estimation and detection of the mth interferer is presented below.

Interference Parameter Estimation
The on-line stochastic gradient algorithm studied in [9] is applied to the posed problem. The model parameters are updated according to the following iterative scheme

$$\theta_b^m = \theta_{b-1}^m + \frac{1-\rho}{1-\rho^{b-1}} K_b^m I_b^m, \tag{22}$$

where ρ is a predetermined real constant and

$$I_b^m = \rho I_{b-1}^m + V_b^m\left(\theta_{b-1}^m\right), \; I_1^m = V_1^m\left(\theta_0^m\right), \; I_0 = 0. \tag{23}$$

and θ_0^m is the initial model parameter estimate. The incremental score vector, V_b^m, is defined by

$$V_b^m\left(\theta_{b-1}^m\right) \triangleq \left. \frac{\partial E\left\{\ln f(Y_b^m, A_b^m | \mathcal{Y}_{b-1}^m, \mathcal{A}_{b-1}^m, \theta^m\right\}}{\partial \theta^m} \right|_{\theta^m=\theta_{b-1}^m} . \tag{24}$$

K_b^m denotes the step size at the bth time instant and is chosen as $K_b^m = 1/b$.
I. The Markov chain parameters.

By using a differential geometric approach suggested in [7], the formulae for updating the transition probabilities are given by the following set of equations

$$\pi_{ij,b}^m = \frac{g_{ij,b}^m}{\sum_{i,j=1}^2 \left(g_{ij,b}^m\right)^2}, \tag{25}$$

where $\pi_{ij,b}^m$ denotes the transition probability estimate on the bth iteration, and

$$g_{ij,b}^m = g_{ij,b-1}^m + \frac{1-\rho}{1-\rho^{b-1}} K_b^m I_b^m, \tag{26}$$

$$I_b^m = \rho I_{b-1} + V_b^m\left(g_{ij,b-1}^m\right), \tag{27}$$

and

$$V_b^m\left(g_{ij,b-1}^m\right) = 2\left(\frac{\alpha_{b-1}^m(i,j)}{g_{ij,b-1}^m} - \alpha_{b-1}^m(i) g_{ij,b-1}^m\right). \tag{28}$$

II. Update of the coefficient, q_2^m.

The element of the incremental score vector corresponding to the state level q_2^m, is given by

$$V_b^m\left(q_{2,b-1}^m\right) = (Y_b^m - NT_s q_{2,b-1}^m)\frac{NT_s}{\sigma^2}\alpha_b^m(2), \tag{29}$$

where $q_{2,b}^m$ denotes the signal strength estimate of the mth interferer on the bth iteration. $\alpha_b^m(2)$ denotes the un-normalized aposteriori probability that the interferer is present on the bth iteration.

Detection (State Estimation)

The minimum probability of error in detecting the mth interferer using data up to time index b is given by

$$A_b^m \;=\; \arg\max_{A_b^m} f(A_b^m \mid \mathcal{Y}_b^m, \theta_b^m) = \arg\max_j \alpha_b^m(j). \tag{30}$$

Interference Suppression

The estimates α^m and q_2^m are used to estimate the mth interferer. The estimate of the narrowband interference at frequency mF_s on the bth data batch is given by the conditional mean estimate

$$\hat{X}_b^m \triangleq E\{A_b^m | \mathcal{Y}_b^m, \theta_b^m\} = q_{2,b}^m \alpha_b^m(2), \tag{31}$$

where $\hat{X}_b^m$ denotes the estimate of X_b^m.

Finally, prior to taking the IDFT, the data Y_b^m in each frequency bin is reduced by the interfering estimate $\hat{X}_b^m$, that is

$$Y_b^m - q_{2,b}^m \alpha_b^m(2) \tag{32}$$

4 Computational Complexity

As stated previously, there is a major advantage to perform the detection, estimation and suppression of the broadband interference in the frequency domain instead of in the time domain. This is so, since the interferers are assumed to be independent in the frequency domain. Our schemes have computational complexities that are linear in the number of interferers instead of exponentially growing as would have been the case if we would have worked in the time domain.

In Table 1, the computational complexities of our off-line and on-line broadband interference schemes are given. Here, the complexity requirements for the Fourier transformation is not included. We choose to perform the transformation using the fast Fourier transform (FFT) which has a computational complexity of $N\log_2(N)$ per batch. Furthermore, the complexity requirements for subtracting the interference for both the on-line and off-line case is $O(M)$ per batch.

	Off-line complexity per pass	*On-line complexity per batch*
γ_b^m	$O(M2^2T)$	$O(M2^2)$
Π^m	$O(M2^2T)$	$O(M2^2)$
q_2^m	$O(MT)$	$O(M)$

Table 1: Computational complexity requirements of the algorithms.

5 Experimental Results

In this section, we illustrate the performance of our proposed off-line and on-line schemes by way of simulation studies. We investigated the performance of our algorithms by varying the level of the amplitudes of the interfering signals and by varying the number of possible narrowband interferers that the broadband interference contains. The off-line and on-line schemes were evaluated for the case when the state level q_2^m is known and the case when the interference amplitude is unknown and has to be estimated. As a comparison to our proposed schemes, we ran a bank of least mean squares (LMS) algorithms in the frequency domain, that replaces our bank of HMM smoothers/filters. The LMS algorithm for the mth interferer is given by

$$C_{b+1}^m = C_b^m + \mu(Y_b^m - C_b^m), \tag{33}$$

where $\mu \geq 0$. Suppression was computed by subtracting C_b^m from Y_b^m prior to taking the IDFT. Note that Eq. (33) reduces to our proposed online interference estimation scheme of (22) and (29) by setting $\rho = 0$, $K_b^m = \mu N T_s/\sigma^2$ and $\alpha_b^m(2) = 1$.

We evaluated the results in the time-domain and we used the SNR improvement as the performance measure. The SNR improvement is defined as

$$SNR_{imp} \triangleq \frac{E(|y_k - s_k|^2)}{E(|y_k - s_k - \hat{x}_k|^2)}, \tag{34}$$

where $\hat{x}_k$ denotes the estimate of the broad-band interference.

Common fixed parameters for all the simulations below were the following. We set the standard deviation of the additive observation noise to be $\sigma_w = 0.001$, the frequency $F = 30000$ and the transition probability matrix Π^m, was given by

$$\Pi^m = \begin{pmatrix} 0.99 & 0.01 \\ 0.01 & 0.99 \end{pmatrix}, \ \forall m = 1, \ldots M. \tag{35}$$

For the off-line case, the number of data batches was $T = 500$ and for the on-line case $T = 1600$. The update constant μ for the LMS algorithm was chosen as 0.05. Fine-tuning of the on-line parameters was not thoroughly investigated. For all the simulation studies below, we ran 100 Monte Carlo simulations. The last 100 data points of each Monte Carlo run were used to compute the SNR improvement.

- *Varying the amplitudes, $q^m(2)$, of the interfering signals.*

The number of possible narrowband interferers were chosen as $M = 16$ and there was only one user transmitting a random sequence of ±1. We varied the amplitudes $q^m(2)$, $m = 1, \dots M$ of the interferers between 0.4 and 1.4. The amplitudes $q^m(2)$ for different $m \in \{1, \dots M\}$ were kept the same in each simulation. This corresponds to varying the SNR between -9.06 dB and -1.81 dB. Firstly, we ran the off-line scheme with known state-levels q_2^m, $m = 1, \dots M$, and thereafter, we ran our off-line schemes assuming unknown state-levels q_2^m, $m = 1, \dots M$. The EM algorithm was iterated five times. The initialization of q_2^m was then set to a random non-zero number. Next, the on-line scheme assuming known state-levels q_2^m, $m = 1, \dots M$, was studied. Thereafter, the on-line scheme assuming unknown state-level q_2^m, $m = 1, \dots M$ was run. The initialization of q_2^m was set to a random non-zero number. Finally, as a comparison a bank of LMS schemes was run under the same conditions. In Fig. 3, the results are presented.

- *Varying the number of possible narrowband interferers.*

The number of users transmitting a random sequence of ±1 was chosen to 1 and the amplitudes $q^m(2)$, $m = 1, \dots M$ of the interferers were all chosen to $q^m(2) = 1$, $m = 1, \dots M$. The number of possible narrowband interferers was varied between 2 and 32. Firstly, we ran the off-line scheme with known state-levels q_2^m, $m = 1, \dots M$, and thereafter, we ran our off-line schemes assuming unknown state-levels q_2^m, $m = 1, \dots M$. The EM algorithm was iterated five times. The initialization of q_2^m was then set to a random non-zero number. Next, the on-line scheme assuming known state-levels q_2^m, $m = 1, \dots M$, was studied. Thereafter, the on-line scheme assuming unknown state-level q_2^m, $m = 1, \dots M$ was run. The initialization of q_2^m was set to a random non-zero number. Finally, as a comparison a bank of LMS schemes was run under the same conditions. In Fig. 4, the results are presented.

As can be seen from Figs. 3–4, the off-line schemes perform better than the on-line schemes, which is not surprising since off-line schemes smooth the data while on-line schemes filter the data. To bridge the gap between the on-line and off-line schemes, fixed-lag smoothers can be used. The parameter estimation via the off-line schemes outperforms the parameter estimation of the on-line schemes. We are currently investigating the efficiency of our parameter estimators in terms of Cramer Rao lower bounds (CRLB).

References

[1] L.B. Milstein, "Interference rejection techniques in spread spectrum communications", *Proc. IEEE*, Vol.76, No.6, pp. 657-671, 1988.

[2] J.W. Ketchum, J. Proakis, "Adaptive algorithms for estimating and suppressing narrow-band interference in spread-spectrum systems", *IEEE Trans. Comms.*, Vol. 30, No. 5, pp. 913–924, 1982.

[3] H.V. Poor and L.A. Rusch, "Narrowband interference suppression in spread spectrum CDMA", *IEEE Personal Comms.*, Third Quarter 1994.

[4] G.J. Saulnier, "Suppression of narrowband jammers in a spread-spectrum receiver using transform-domain adaptive filtering", *IEEE Jnl. on Selected Areas in Comms.*, Vol. 10, No. 4, pp. 742–749, 1992.

[5] L.M. Li and L.B. Milstein, "Rejection of pulsed CW interference in PN spread-spectrum systems Using complex adaptive filters", *IEEE Trans. on Comms.*, Vol.31, No.1, pp. 10-20, January 1983.

[6] C. F. J. Wu, "On the convergence properties of the EM algorithm", *The Annals of Statistics*, Vol. 11, No. 1, pp. 95–103.

[7] I. B. Collings, V. Krishnamurthy, J. B. Moore, "On-Line Identification of Hidden Markov Models via Recursive Prediction Error Techniques", *IEEE Trans. on Signal Proc.*, Vol. 42, No. 12, pp. 3535–3539, Dec. 1994.

[8] L. R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition", *Proc. IEEE*, Vol. 77, pp. 257-286, 1989.

[9] E. Weinstein, A.V. Oppenheim, M. Feder and J.R. Buck, *Iterative and Sequential Algorithms for Multisensor Signal Enhancement*, IEEE Trans. on Signal Proc., Vol.42, No.4, pp 846-859, April 1994.

[10] S. Dey, V. Krishnamurthy and T. Salmon-Legagneur, "Estimation of Markov-modulated time-series via EM algorithm", *IEEE Signal Proc. Letters*, vol. 1, no. 10, pp. 153–155, 1994.

[11] A. P. Dempster, N. M. Laird and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm", *Jnl. of Royal Statistical Society*, vol. 39, pp. 1–38, 1977.

[12] C. Carlemalm, H.V. Poor, A. Logothetis, "Broadband Interference Suppression in a Spread-spectrum Communication System", submitted to *IEEE Jnl. on Selected Areas in Comms.*, 1999.

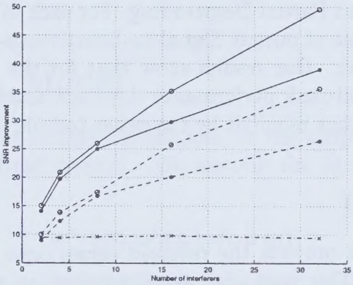


Figure 2: Performance in terms of SNR improvement for varying the amplitudes, $q^m(2)$, of the interfering signals. Off-line scheme results are denoted by solid lines '–', on-line scheme results by dashed lines '- -' and finally LMS algorithm results by dashed-dotted line '-.-'. We use 'o' and '*' to show the performance when the parameters are known and unknown, respectively.

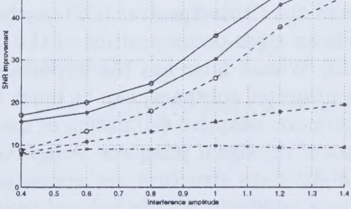


Figure 3: Performance in terms of SNR improvement for varying number of possible interferers. The same notation as in Fig. 3 holds.

Large System Performance of Reduced-Rank Linear Filters

Michael L. Honig and Weimin Xiao

Department of ECE, Northwestern University
2145 Sheridan Road, Evanston, IL 60208

Abstract

The performance of reduced-rank interference suppression is studied for Direct-Sequence (DS)-Code Division Multiple Access (CDMA). A reduced-rank filter resides in a lower-dimensional space, relative to the full-rank filter, and therefore requires fewer data samples to estimate the filter coefficients in an adaptive mode. We consider the following reduced-rank techniques: the Multi-stage (MS) Wiener filter of Goldstein and Reed, Principal Components and Cross-Spectral (based on an eigen-decomposition of the covariance matrix), and Partial Despreading. For each of these techniques we evaluate the large system limit of output Signal-to-Interference-Plus-Noise-Ratio (SINR) as a function of filter rank. Our results show that for the MS filter, the filter rank needed to achieve near full-rank performance does not scale with system size, and is typically much less than that required by the other techniques.

1 Introduction

A reduced-rank filter first projects the received signal onto a lower-dimensional subspace before processing. This type of dimension reduction can improve tracking and convergence in time-varying environments. Reduced-rank linear filtering has been studied primarily for array processing and radar applications; however, recently it has been proposed for interference suppression in Direct-Sequence (DS)-Code-Division Multiple Access (CDMA) systems [1, 2].

We compare the performance of reduced-rank interference suppression techniques for DS-CDMA based on a Minimum Mean Squared Error (MMSE) performance criterion. The techniques considered are: (1) the Multi-stage Wiener filter of Goldstein and Reed [3], (2) Principal Components [1, 2], (3) Cross-Spectral [4], and (4) Partial-Despreading [5]. Principal Components (PC) is perhaps the most widely known reduced-rank method, and projects the received signal onto the subspace with largest energy. Both PC and the Cross-Spectral (CS) method require an estimate of the signal subspace, e.g., via an eigen-decomposition of the input covariance matrix.

The multi-stage (MS) Wiener filter has the important property that the filter rank (i.e., dimension of the projected subspace) can be much less than the dimension of the signal subspace. Furthermore, adaptive estimation of the optimal filter coefficients does not require an estimate of the signal subspace. Adaptive interference suppression algorithms based on the multi-stage structure are presented in [6]. Finally, Partial Despreading (PD) refers to a relatively simple reduced-rank technique in which the subspace is spanned by nonoverlapping segments of the matched filter. (See also [7] for a comparison of PC, CS, and PD techniques.)

For each of the preceding methods, we compute the output Signal-to-Interference Plus Noise Ratio (SINR) as the number of users, K, and processing gain $N \rightarrow \infty$ with

K/N fixed. This type of large system analysis with random signature sequences was presented in [8] for the MMSE linear receiver. For the MS filter, assuming all users are received with the same power, we obtain a closed-form expression for the output SINR as a function of filter rank D, K/N, and the background noise variance. As D increases, this expression rapidly converges to the full-rank large-system SINR derived in [8] as a *continued-fraction*. The MS filter therefore has the remarkable property that the dimension D needed to obtain a target SINR (e.g., within a small ϵ of the full-rank SINR) does not scale with the system size (i.e., K and N). Our results show that for moderate to heavy loads, $D = 8$ is sufficient to achieve essentially full-rank performance, and $D = 4$ is within a fraction of a dB of the full-rank SINR.

For the other reduced-rank methods considered, we define the large system limit as $D, K, N \to \infty$ with both K/N and D/K fixed. That is, to achieve a target SINR near the large-system limit, $D \to \infty$ as $K, N \to \infty$. For the case where all users are received with the same power, we obtain expressions for output SINR as a function of K/N, D/K, and noise variance. A comparison with simulation results shows that the large system results accurately predict the performance of a moderately sized system (e.g., $N = 32$).

2 DS-CDMA Model

Let $\mathbf{r}(i)$ be the N-vector containing samples at the output of a chip-matched filter during the ith transmitted symbol. We assume the idealized synchronous CDMA model

$$\mathbf{r}(i) = \mathbf{S}\mathbf{A}\mathbf{b}(i) + \mathbf{n}(i) \tag{1}$$

where $\mathbf{S} = [\mathbf{s}_1, \ldots, \mathbf{s}_K]$ is the $N \times K$ matrix of spreading codes and $\mathbf{s}_k$ is the spreading code associated with user k, $\mathbf{A}$ is the diagonal matrix of amplitudes, $\mathbf{b}(i)$ is the K-vector of symbols across users at time i, and $\mathbf{n}(i)$ is the noise vector, which has covariance matrix $\sigma^2\mathbf{I}$. We assume that the symbol variance is one for all users.

In what follows we assume that user 1 is the desired user. The Minimum Mean Squared Error (MMSE) receiver consists of the vector $\mathbf{c}$, which is chosen to minimize the MSE

$$\mathcal{M} = E\{|b_1(i) - \mathbf{c}^\dagger \mathbf{r}(i)|^2\} \tag{2}$$

where † represents Hermitian transpose. The MMSE solution is

$$\mathbf{c} = \mathbf{R}^{-1}\mathbf{s}_1 \tag{3}$$

where the $N \times N$ covariance matrix

$$\mathbf{R} = E[\mathbf{r}(i)\mathbf{r}^\dagger(i)] = \mathbf{S}\mathbf{A}\mathbf{A}^\dagger\mathbf{S}^\dagger + \sigma^2\mathbf{I} \tag{4}$$

Let

$$\mathbf{S}_I = [\mathbf{s}_2, \ldots, \mathbf{s}_K], \tag{5}$$

that is, the $N \times (K-1)$ matrix of spreading codes for the interferers,

$$\mathbf{r}_I(i) = \mathbf{S}_I\mathbf{A}_I\mathbf{b}_{2:K}(i) + \mathbf{n}(i) \tag{6}$$

where $\mathbf{A}_I$ is the diagonal matrix of interference amplitudes and $\mathbf{x}_{l:m}$ denotes components l through m of the vector $\mathbf{x}$, and

$$\mathbf{R}_I = E[\mathbf{r}_I(i)\mathbf{r}_I^\dagger(i)] = \mathbf{S}_I\mathbf{A}_I\mathbf{A}_I^\dagger\mathbf{S}_I^\dagger + \sigma^2\mathbf{I} \tag{7}$$

That is, $\mathbf{R}_I$ is the interference-plus-noise covariance matrix. Then the output Signal-to-Interference-Plus Noise Ratio (SINR) of the MMSE filter is

$$\beta = \mathbf{s}_1^\dagger\mathbf{R}_I^{-1}\mathbf{s}_1 \tag{8}$$

3 Reduced-Rank Linear Filtering

Let $\mathbf{M}_D$ be the $N \times D$ matrix with column vectors which are an orthonormal basis for a D-dimensional subspace, where $D < N$. The projected received vector corresponding to symbol i is then

$$\tilde{\mathbf{r}}(i) = \mathbf{M}_D^\dagger\mathbf{r}(i), \tag{9}$$

where, in what follows, all D-dimensional (projected) quantities are denoted with a "tilde".

The sequence of projected received vectors $\{\tilde{\mathbf{r}}(i)\}$ is the input to a tapped-delay line filter, represented by the D-vector $\tilde{\mathbf{c}}(i)$ for symbol i. The filter output corresponding to the ith transmitted symbol is $z(i) = \tilde{\mathbf{c}}^\dagger(i)\tilde{\mathbf{r}}(i)$, and the objective is to select $\tilde{\mathbf{c}}$ to minimize the reduced-rank MSE $E\{|b_1(i) - \tilde{\mathbf{c}}^\dagger\tilde{\mathbf{r}}(i)|^2\}$. The solution is $\tilde{\mathbf{c}} = \tilde{\mathbf{R}}^{-1}\tilde{\mathbf{s}}_1$ where $\tilde{\mathbf{R}} = \mathbf{M}_D^\dagger\mathbf{R}\mathbf{M}_D$ and $\tilde{\mathbf{s}}_1 = \mathbf{M}_D^\dagger\mathbf{s}_1$, the MMSE is given by

$$\mathcal{M}_D = 1 - \tilde{\mathbf{s}}_1^\dagger\tilde{\mathbf{R}}^{-1}\tilde{\mathbf{s}}_1 \tag{10}$$

and the output SINR is $\beta_D = \tilde{\mathbf{s}}_1^\dagger\tilde{\mathbf{R}_I}^{-1}\tilde{\mathbf{s}}_1$ where $\tilde{\mathbf{R}}_I = \mathbf{M}_D^\dagger\mathbf{R}_I\mathbf{M}_D$.

3.1 Multi-Stage Wiener Filter

A block diagram showing four stages of the multi-stage Wiener filter is shown in Figure 1. The stages are associated with the sequence of nested filters $\mathbf{c}_1, \ldots, \mathbf{c}_D$, where D is the order of the filter. Let $\mathbf{B}_m$ denote a *blocking matrix*, i.e.,

$$\mathbf{B}_m^\dagger\mathbf{c}_m = \mathbf{0}. \tag{11}$$

Referring to Figure 1, let $d_m(i)$ denote the output of the filter $\mathbf{c}_m$, and $\mathbf{r}_m(i)$ denote the output of the blocking matrix $\mathbf{B}_m$, both at time i. Then the $m+1$st multi-stage filter is determined by

$$\mathbf{c}_{m+1} = E[d_m^*\mathbf{r}_m] \tag{12}$$

For $m = 0$, we have $d_0(i) = b_1(i)$ (the desired input symbol), $\mathbf{r}_0(i) = \mathbf{r}(i)$, and $\mathbf{c}_1$ is the matched filter (MF) $\mathbf{s}_1$. As in [3], it will be convenient to normalize the filters $\mathbf{c}_1, \ldots, \mathbf{c}_D$ so that $||\mathbf{c}_i|| = 1$ for each i.

The filter output is obtained by linearly combining the outputs of the filters $\mathbf{c}_1, \ldots, \mathbf{c}_D$ via the weights $w_1, \ldots, w_{D-1}$. This is accomplished stage-by-stage as shown in Figure 1. The MS filter is given by the following set of recursions.

Initialization:

$$d_0(i) = b_1(i) \qquad \mathbf{r}_0(i) = \mathbf{r}(i) \tag{13}$$

For $n = 1, \ldots, D$ (*Forward Recursion*):

$$\begin{aligned} \mathbf{c}_n &= E[d_{n-1}\mathbf{r}_{n-1}(i)]/\|Ed_{n-1}\mathbf{r}_{n-1}\| & (14)\\ d_n(i) &= \mathbf{c}_n^\dagger \mathbf{r}_{n-1}(i) & (15)\\ \mathbf{B}_n &= \mathbf{I} - \mathbf{c}_n\mathbf{c}_n^\dagger & (16)\\ \mathbf{r}_n &= \mathbf{B}_n^\dagger \mathbf{r}_{n-1} & (17) \end{aligned}$$

Decrement $n = D, \ldots, 1$ (*Backward Recursion*):

$$\begin{aligned} w_n &= E[d_{n-1}(i)\epsilon_n(i)]/E[|\epsilon_n(i)|^2] & (18)\\ \epsilon_{n-1}(i) &= d_{n-1}(i) - w_n^* \epsilon_n(i) & (19) \end{aligned}$$

where $\epsilon_D(i) = d_D(i)$.

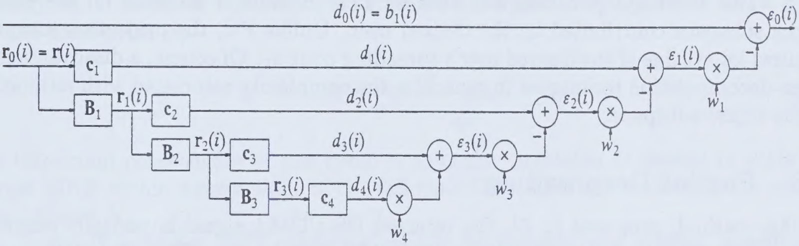

$d_0(i) = b_1(i)$
$\varepsilon_0(i)$
$\mathbf{r}_0(i) = \mathbf{r}(i)$
$\mathbf{c}_1$
$d_1(i)$
$\varepsilon_1(i)$
$\mathbf{B}_1$
$\mathbf{r}_1(i)$
$\mathbf{c}_2$
$d_2(i)$
$\varepsilon_2(i)$
w_1
$\mathbf{B}_2$
$\mathbf{r}_2(i)$
$\mathbf{c}_3$
$d_3(i)$
$\varepsilon_3(i)$
w_2
$\mathbf{B}_3$
$\mathbf{r}_3(i)$
$\mathbf{c}_4$
$d_4(i)$
w_3
w_4
+
−
×

Figure 1: Multi-stage Wiener filter.

The MS filter has the following properties:

1. At stage n the filter generates the desired sequence $\{d_n(i)\}$ and the "observation" sequence $\{\mathbf{r}_n(i)\}$. The MS filter is "self-similar" in that the filter for estimating the former from the latter has the same structure at each stage. The full-rank MMSE filter can therefore be represented as an MS filter with n stages, where $\mathbf{c}_n$ is replaced by the MMSE filter for estimating $d_{n-1}(i)$ from $\mathbf{r}_{n-1}(i)$.
2. Computation of the MMSE filter coefficients does not require an estimate of the signal subspace, as do the Eigen-decomposition techniques to be described. Successive filters are determined by "Residual Correlations" of signals in the preceding stage.

3.2 Eigen-Decomposition Methods

The covariance matrix can be written as

$$\mathbf{R} = \mathbf{V}\Lambda\mathbf{V}^\dagger \tag{20}$$

where $\mathbf{V}$ is the orthonormal matrix of eigenvectors of $\mathbf{R}$ and Λ is the diagonal matrix of eigenvalues. Suppose that the eigenvalues are ordered as $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_N$. For given subspace dimension D, the projection matrix for PC, $\mathrm{M}_D = \mathrm{V}_{1:D}$, the first D columns of $\mathbf{V}$.

For $K < N$, we note that eigenvalues $\lambda_1, \ldots, \lambda_K$ are associated with the signal subspace, and the remaining eigenvalues are associated with the noise subspace, i.e., $\lambda_m = \sigma^2$ for $K < m \leq N$. Consequently, if $D \geq K$, then PC retains the full-rank MMSE performance. However, the performance can degrade quite rapidly for $D \leq K$, since there is no guarantee that the associated subspace will retain most of the desired signal energy.

The CS technique chooses M_D as the set of D eigenvectors which minimizes the MSE. If Λ_D is the associated $D \times D$ matrix of eigenvalues, then from (10) the MSE is

$$\mathcal{M}_{\mathcal{D}} = 1 - ||\Lambda_D^{-1}\tilde{\mathbf{s}}_1||^2 \tag{21}$$

The subspace that minimizes the MSE has basis vectors which are the eigenvectors of $\mathbf{R}$ associated with the D largest values of $|\tilde{\mathbf{s}}_{1,k}/\lambda_k|^2$, where $\tilde{\mathbf{s}}_{1,k}$ is the kth component of $\tilde{\mathbf{s}}_1$, and is given by $\mathbf{V}_k^\dagger \mathbf{s}_1$, where $\mathbf{V}_k$ is the kth column of $\mathbf{V}$. (Note the inverse weighting of $|\lambda_k|^2$ in contrast with PC.)

This technique performs well even for $D < K$ since it accounts for the energy in the subspace contributed by the desired user. Unlike PC, the projection subspace requires knowledge of the desired user's spreading code $\mathbf{s}_1$. Of course, a disadvantage of eigen-decomposition techniques in general is the complexity associated with estimation of the signal subspace.

3.3 Partial Despreading

In this method, proposed in [5], the received DS-CDMA signal is partially despread over consecutive segments of M chips, where M is a parameter. The partially despread vector has dimension $D = \lceil N/M \rceil$, and is the input to the D-tap filter. Consequently, $M = 1$ corresponds to the full-rank MMSE filter, and $M = N$ corresponds to the MF. The columns of M_D in this case are nonoverlapping segments of $\mathbf{s}_1$, where each segment is of length M.

Specifically, if N/M is an integer,

$$[\mathrm{M}_D]_j{}' = \sqrt{N/M}[0 \ldots 0 \ \mathbf{s}_{1,[(j-1)M+1:jM]}{}' \ 0 \ldots 0] \tag{22}$$

where $1 \leq j \leq D$, and there are $(M-1)M$ zeros on the left and $(N/M - j)M$ zeros on the right. This is a simple reduced-rank technique that allows the selection of MMSE performance between the matched and MMSE filters by increasing the number of adaptive filter coefficients.

4 Main Results

The large system limits of the output SINR for the reduced-rank techniques presented in the last section are given by the following Theorems. Proofs are given in the associated full paper [9]. For finite K and N the output SINR is a random variable due to the assignment of random signature sequences. In analogy with the full-rank MMSE receiver, the output SINR for each of the reduced-rank techniques considered converges to a deterministic quantity as $K = \alpha N \rightarrow \infty$.

4.1 Multi-Stage Filter

We first assume that all users are received with the same power.

Theorem 1 *As $K = \alpha N \to \infty$, the output SINR of the rank D multi-stage filter converges in probability to the limit β_D^{MS}, which satisfies*

$$\beta_{D+1}^{MS} = \frac{P}{\sigma^2 + \alpha \frac{P}{1+\beta_D^{MS}}} \quad \textit{for } D \geq 0 \tag{23}$$

where P is the received power for each user, $\beta_0^{MS} = 0$, and $\beta_1^{MS} = P/(\sigma^2 + \alpha P)$ is the large system limit of the output SINR for the matched filter.

According to this theorem, for $D < N$ the output SINR of the MS filter can be expressed as a *continued fraction.* For example,

$$\beta_2^{MS} = \frac{P}{\sigma^2 + \alpha \frac{P}{1+\frac{P}{\sigma^2+\alpha P}}} \tag{24}$$

Clearly, as D increases this continued fraction converges to the full-rank SINR given by the fixed-point [8]

$$\beta^* = \frac{P}{\sigma^2 + \alpha \frac{P}{1+\beta^*}}. \tag{25}$$

An important consequence of this result is that the dimension D needed to achieve a target SINR within some small ϵ of the full-rank SINR does not scale with the system size (K and N).

As D increases, β_D^{MS} converges rapidly to the full-rank MMSE. Specifically, consider the case without background noise, $\sigma^2 \to 0$. It is easy to show that

$$\beta_D^{MS} = \sum_{n=1}^{D} \alpha^{-n} = \begin{cases} \frac{\alpha^{-i}-1}{1-\alpha} & \text{for } \alpha < 1 \\ D & \text{for } \alpha = 1 \\ \frac{1-\alpha^{-D+1}}{1-\alpha} & \text{for } \alpha > 1 \end{cases} \tag{26}$$

In particular, β_D^{MS} increases exponentially with D for $\alpha < 1$ and linearly with D for $\alpha = 1$. If $\alpha > 1$, then the gap between β_i^{MS} and the full-rank performance $\beta^* = \frac{1}{\alpha-1}$ decreases exponentially.

We now consider the MS filter with an arbitrary power distribution. In this case we do not have a closed-form result for the output SINR, although we can compute the output SINR numerically. In analogy with the uniform power case we also have the *approximation*

$$\beta_{D+1}^{MS} \approx \frac{P_1}{\sigma^2 + \alpha \int_0^\infty \frac{P_1 P}{P_1 + P\beta_D^{MS}} dF(P)} \tag{27}$$

where $\beta_0^{MS} = 0$, $\beta_1^{(MS)} = P/\left(\sigma^2 + \alpha \int_0^\infty P dF(P)\right)$ is the asymptotic MF SINR, and $F(P)$ is an arbitrary power distribution. As $D \to \infty$, equality holds [8]; however, it is shown in [9] that this expression is not exact for finite D.

To compute the output SINR for the MS filter with an arbitrary power distribution, we first give an alternate representation of the subspace spanned by the columns of the projection matrix $\mathbf{M}_D$. Let $\mathcal{S}_D$ denote the D-dimensional subspace associated with the rank D MS filter, and let $\mathbf{R}_I$ denote the interference plus noise covariance matrix given by (7).

Lemma 1: The subspace $\mathcal{S}_D$ is spanned by the vectors $\mathbf{v}_0, \ldots, \mathbf{v}_{D-1}$ where $\mathbf{v}_n = \mathbf{R}_I^n \mathbf{s}_1$.

Let

$$\begin{aligned} \gamma_m &= \mathbf{s}_1^\dagger \mathbf{v}_i = \mathbf{s}_1^\dagger \mathbf{R}_I^m \mathbf{s}_1 && (28) \\ \gamma_{l:m} &= [\gamma_l \ \gamma_{l+1} \cdots \gamma_m]' && (29) \\ \Gamma_{l:l+m} &= [\gamma_{l:l+m} \ \gamma_{l+1:l+m+1} \cdots \gamma_{l+m:l+2m}] && (30) \end{aligned}$$

Note that $\Gamma_{l:l+m}$ is an $(m+1) \times (m+1)$ matrix.

To compute the large system SINR, we need to compute the large system limit of γ_n, $0 \le n \le 2D-1$, which is given by the following Lemma, where G is the asymptotic eigenvalue distribution of $\mathbf{S}_I \mathbf{A} \mathbf{A}^\dagger \mathbf{S}_I^\dagger$ [10].

Lemma 2 *As $K = \alpha N \to \infty$, γ_n converges in probability to the limit*

$$\gamma_n^* = \int (\lambda + \sigma^2)^n dG(\lambda) \qquad (31)$$

We can now state the large system SINR of the rank D MS filter. The superscript $*$ indicates the large system limit of the associated variable(s).

Theorem 2 *As $K = \alpha N \to \infty$, the output SINR of the MS filter given an arbitrary power distribution F converges in probability to*

$$\beta_D^{(MS)} = (\gamma_{0:D-1}^*)^\dagger (\Gamma_{1:D}^*)^{-1} \gamma_{0:D-1}^* \qquad (32)$$

As for the uniform power case, the dimension D needed to achieve a target SINR within some small constant ϵ of the full-rank SINR does not scale with K and N.

For large D Theorem 2 gives an alternative method to that in [8] for computing the full-rank MMSE. For the uniform power case, (32) must be equivalent to (23). We also remark that it is possible to compute the sequence γ_k without explicit use of the asymptotic eigenvalue distribution G [9].

4.2 Eigen-Decomposition Methods

In what follows, the large system limit is defined as $D, K, N \to \infty$ where $K/N = \alpha$ and $D/N = \delta$. In particular, D now increases proportionally with K and N. For PC we only discuss the large system limit for the uniform power case. The output SINR can be written as

$$\beta_D = \frac{v_D}{1 - v_D} \qquad (33)$$

where

$$v_D = P \sum_{n=1}^{D} \frac{|\mathbf{V}_n^\dagger \mathbf{s}_1|^2}{\lambda_n + \sigma^2} \qquad (34)$$

where λ_n and $\mathbf{V}_n$ are the nth eigenvalue and eigenvector of the covariance matrix $\mathbf{R}$, and $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_N$.

We can evaluate

$$\lim_{K=\alpha N \to \infty} E\left[v(\alpha, \delta)\right] = \frac{1}{\alpha} \int_c^{b(\alpha)} \frac{\lambda}{\lambda + \sigma^2} dG(\lambda) \tag{35}$$

where c is defined by $G(c) = 1 - \delta$. This integral and $G(c)$ can be evaluated in closed-form [9]. We conjecture that as $K = \alpha N \to \infty$, $v(\alpha, \delta)$ converges to the preceding limit in probability, in which case β_D converges in probability to the corresponding expression (33). The main difficulty with proving this is that $\mathbf{V}_n$ and $\mathbf{s}_1$ are correlated.

Although we do not have an analogous result for an arbitrary power distribution, we observe that for $\delta \geq \alpha$ the PC algorithm again achieves full-rank performance, since the eigenvectors chosen for the projection include the signal subspace. As δ decreases below α, in a near-far situation the performance can be substantially worse than the uniform power case since the subspace $\mathcal{S}_{\mathcal{D}}$ is likely to be dominated by the interferers.

For the Cross-Spectral method (33) and (34) again apply with a different ordering for eigenvalues/eigenvectors, and we have that

$$\lim_{K=\alpha N \to \infty} E\left[v_D(\alpha, \delta)\right] = 2 \int_c^{\infty} |\xi|^2 dH(\xi) \tag{36}$$

where $H(\xi)$ is the empirical distribution of $\xi = \frac{\mathbf{V}_n^\dagger \mathbf{s}_1}{\sqrt{\lambda_n + \sigma^2}}$ and c satisfies $H(c) = 1 - \frac{\delta}{2}$. We do not have an explicit expression for $H(\cdot)$. However, a reasonable assumption is that $H(\cdot)$ is Gaussian. The variance of ξ is $\frac{\beta^*}{1+\beta^*}$ where β^* is the full-rank SINR. Once again, we conjecture that $v_D(\alpha, \delta)$ converges to the large system limit in probability, in which case β_D converges in probability to the corresponding limit given by (33). The numerical results shown in the next section are generated according to these assumptions.

4.3 Partial Despreading (PD)

The output SINR for Partial Despreading is expressed in terms of the full-rank asymptotic SINR, which we denote as $\mathcal{B}_F[\alpha, P_1, \sigma^2, F(\cdot)]$ (i.e., β^* from (27) for $D \to \infty$). The large system limit is obtained by despreading over $M \geq 1$ chips, where M is held constant, so that $N/D = M = 1/\delta$.

Theorem 3 *As $K = \alpha N \to \infty$ and $D = N/M \to \infty$, the output SINR of the Partial Despreading (PD) MMSE receiver converges in probability to the limit*

$$\beta^{PD}(\alpha, M) = \mathcal{B}_F\left[M\alpha, MP_1, M\sigma^2, F(\cdot)\right] \tag{37}$$

where $M = 1/\delta$.

If $\delta = 0$ ($M \to \infty$), then β^{PD} is the large system limit for the MF SINR.

5 Numerical Results

Figure 2 shows output SINR vs. normalized rank D/K for the reduced-rank filters considered assuming uniform power, $SNR = 10$ dB, and $\alpha = 1/2$. Also shown are simulation results for $N = 32$ averaged over spreading codes. For all four methods considered, the large system analysis accurately predicts the simulation results. As discussed previously, the asymptotic SINR for the MS filter shown in the Figure is the full-rank SINR for all D/K, since (essentially) full-rank SINR is achieved for finite D, independent of K and N. For the MS filter both analytical and simulation results are shown for $D = 1, 2, 4$, and 8.

There is a large gap between the results for the MS filter and the other methods for small D/N. The CS and PC reduced-rank filters can achieve full-rank performance only when $D \geq K$. For $D < K$, these results show that the CS filter performs much better than the PC filter. The Partial Despreading filter can achieve the full-rank performance only when $D = N$, since for any $D < N$, the selected subspace $\mathcal{S}_D$ does not generally contain the MMSE solution. For small D/K the PD filter performs close to the Matched Filter, which is significantly better than the Eigen-decomposition methods.

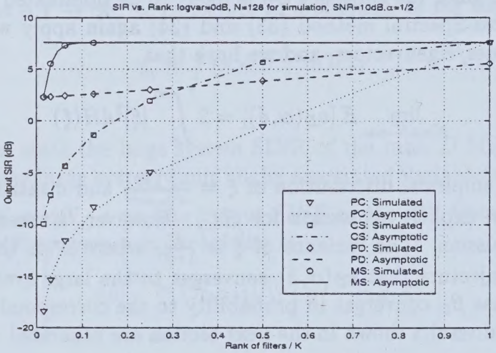


Figure 2: Output SINR for different reduced rank filters: Uniform Power

6 Conclusions

Explicit expressions for large system output SINR have been obtained for four different reduced-rank filters. An important conclusion from the analysis of the MS filter is that the D needed to achieve near full-rank performance does not scale with K and N. Numerical results for $\alpha = 1/2$ indicate that $D = 8$ is sufficient to achieve near full-rank performance. Further computations show that this is true even for very heavy loads (e.g., $\alpha = 1$). In contrast, for the other techniques considered, D must increase proportionally with K or N to achieve similar performance.

For very small D/K, the techniques based on eigen-decomposition perform relatively poorly. In particular, The cross-spectral filter requires $D/K > 0.25$ to perform better than the matched filter. The PC method performs well only when $D/K \geq 1$.

Our results pertain to optimal (MMSE) filters. Adaptive "Residual Correlation (ResCor)" algorithms based on the multi-stage filter were presented in [6], and were shown to perform significantly better than other reduced-rank adaptive filters. Analytical characterization of the convergence properties of these adaptive ResCor filters is currently being pursued.

References

[1] X. Wang and H.V. Poor. Blind multiuser detection: A subspace approach. *IEEE Trans. Inform. Theory*, 44(2):677–690, March 1998.

[2] M. L. Honig. Adaptive linear interference suppression for packet DS-CDMA. *European Transactions on Telecommunications*, 9(2):173–182, March-April 1998.

[3] J.S. Goldstein, I.S. Reed, and L.L. Scharf. A multistage representation of the Wiener filter based on orthogonal projections. *IEEE Trans. Inform. Theory*, 44(7), November 1998.

[4] J.S. Goldstein and I.S. Reed. Reduced rank adaptive filtering. *IEEE Trans. Signal Processing*, 45(2):492–496, February 1997.

[5] R. Singh and L.B. Milstein. Adaptive interference suppression in direct-sequence spread-spectrum CDMA. *Submitted to IEEE Trans. Commun.*

[6] M. L. Honig and J. S. Goldstein. Adaptive reduced-rank residual correlation algorithms for DS-CDMA interference suppression. In *Proc. 32nd Asilomar Conf. Signals, Syst. Comput.*, Pacific Grove, CA, November 1998.

[7] M.L. Honig. A comparison of subspace adaptive filtering techniques for DS-CDMA interference suppression. In *Proc. IEEE MILCOM*, volume 2, pages 836–840, Monterey, CA, November 1997.

[8] D. N. C. Tse and S. V. Hanly. “Linear Multiuser Receivers: Effective Interference, Effective Bandwidth and User Capacity”. *IEEE Trans. Inform. Theory*, 45(2):641–657, March 1999.

[9] W. Xiao and M. L. Honig. Large system performance of reduced-rank linear filters. in preparation, 1999.

[10] J. W. Silverstein and Z. D. Bai. “On the Empirical Distribution of Eigenvalues of a Class of Large Dim ensional Random Matrices”. *J. Multivariate Anal.*, 54(2):175–192, 1995.

Nonlinear Multiuser Receivers with Distributed Power Control in Cellular Radio Networks

Mahesh K. Varanasi
Department of Electrical and Computer Engineering, University of Colorado
Boulder, CO USA 80309-0425

Abstract

Uplink communication in a cellular radio network is considered where the base stations in each cell employ a linear or nonlinear (decision feedback) multiuser receiver (cf. [1]). For any such receiver, the problem of minimizing total transmit power is considered under the constraint that all the users of the network achieve their Quality-of-Service (QoS) objective in terms of signal-to-interference ratio (SIR). A distributed power control algorithm based on the stochastic approximation method is proposed that converges in the mean-square sense to the minimal powers when the target SIRs are feasible. The feasibility region is much larger for a well-designed multiuser receiver, particularly for a nonlinear receiver that employs decision feedback, than it is for the linear single-user receiver. The problem of further minimizing the minimum total transmit power over the feedforward and feedback filters of the base station receivers is also solved. A distributed algorithm is obtained that converges to the jointly optimal pair of power allocations and (maximum SIR) decision feedback receivers.

I. INTRODUCTION AND SYSTEM MODEL

A combined approach to multiuser receivers and power control was proposed in [2] [3] and [4]. In [3], it was shown that nonlinear multiuser receivers have the potential for significantly higher performance than do their linear counterparts. Those results were obtained for a single-cell channel and deterministic power control. This work considers decentralized, iterative, and stochastic power control for a cellular radio network where the base stations employ multiuser receivers with or without decision feedback. Recently, stochastic power control algorithms were proposed for networks in which base stations use linear single-user receivers [5].

In addition to obtaining optimum power control strategies for fixed linear and nonlinear receivers, the problem of minimizing the minimum total transmit power subject to QoS constraints over the feedforward and feedback filters of the base station receivers is also solved. This idea was first proposed for linear receivers by the author in [6] for a single-cell channel and independently for a multi-cell model in [4]. In this paper, we consider the larger class of linear and nonlinear receivers for a multi-cell model thereby obtaining the most power-efficient solution to the problem of achieving a given set of QoS constraints given that the base station receivers are of the linear or decision feedback type [1].

Consider a cellular network in which there are B base stations and K active users with K_j users assigned to base j. Base j employs a set of N_j matched filters matched to orthonormal basis functions of a carefully chosen signal subspace that must include the signals received from the users within its cell. While the transmissions of the mobile users of a particular cell arrive at the bases of other cells symbol-asynchronously, it is assumed, for the sake of simplicity, that they arrive at their own base symbol-synchronously. It is also assumed that while the receiver at base j has knowledge of the common timing of the received signals of its own cell, it does not have the timing information associated with signals of other cells.

The discrete-time model for the N_j matched filter outputs at base j can be expressed as

$$\mathbf{y}_j = \sum_{i=1}^{K_j} \sqrt{w_{ij} g_{ijj}}\, \mathbf{s}_{ijj} x_{ij} + \sum_{l \neq j}^{B} \sum_{i=1}^{K_l} \sqrt{w_{il} g_{ilj}} \left(\mathbf{s}_{ilj}^{-} x_{il}^{-} + \mathbf{s}_{ilj}^{+} x_{il}^{+} \right) + \mathbf{n}_j \,, \qquad (1)$$

where w_{il} and x_{il} denote the transmit power and the transmitted symbol, respectively, of user i of base l. g_{ilj} denotes the channel gain of user i of base l to base j. The vectors $\mathbf{s}_{ijj}$ denote the

---

This work was supported by NSF Grant CCR-9725778 and CCR-9814996.

vector representations (the "signature sequence") of the signal of user i of base j with respect to the orthonormal basis functions employed at base j. The vectors $\mathbf{s}_{ilj}^{-}$ and $\mathbf{s}_{ilj}^{+}$ denote the vector representations of the segments of the signals associated with the two symbols of user i of base l that overlap with the signals of base j relative to the orthonormal basis functions employed at base j. $\mathbf{n}_j$ is an N_j-dimensional zero-mean Gaussian random vector with a covariance matrix equal to $\sigma_j^2\mathbf{I}$. Define the signal matrix $\mathbf{S}_{jj} = \left[\mathbf{s}_{1jj}\mathbf{s}_{2jj}\cdots\mathbf{s}_{K_jjj}\right]$ whose columns are hence the signature sequences of the users of cell j. Also define the matrices $\mathbf{S}_{lj}^{+} = \left[\mathbf{s}_{1lj}^{+}\mathbf{s}_{2lj}^{+}\cdots\mathbf{s}_{K_llj}^{+}\right]$ and $\mathbf{S}_{lj}^{-} = \left[\mathbf{s}_{1lj}^{-}\mathbf{s}_{2lj}^{-}\cdots\mathbf{s}_{K_llj}^{-}\right]$. The received signal can therefore be written as

$$\mathbf{y}_j = \mathbf{S}_{jj}\mathbf{G}_{jj}^{1/2}\mathbf{W}_j^{1/2}\mathbf{x}_j + \sum_{l\neq j}^{B}\mathbf{S}_{lj}^{-}\mathbf{G}_{lj}^{1/2}\mathbf{W}_l^{1/2}\mathbf{x}_l^{-} + \sum_{l\neq j}^{B}\mathbf{S}_{lj}^{+}\mathbf{G}_{lj}^{1/2}\mathbf{W}_l^{1/2}\mathbf{x}_l^{+} + \mathbf{n}_j\,, \tag{2}$$

where $\mathbf{G}_{lj} = \operatorname{diag}\{g_{1lj}, g_{2lj}, \cdots, g_{K_llj}\}$, and $\mathbf{W}_l = \operatorname{diag}\{w_{1l}, w_{2l}, \cdots, w_{K_ll}\}$. Moreover, $\mathbf{x}_j$, $\mathbf{x}_l^{+}$ and $\mathbf{x}_l^{-}$ are vectors whose i^{th} elements are x_{ij}, x_{il}^{+} and x_{il}^{-}, respectively.

II. MULTIUSER RECEIVERS

Each base station uses either a linear or a decision feedback receiver [1] to decode the information transmitted by users in its own cell. In the decision feedback case, assume for notational simplicity, that the users in each cell are numbered according to the order in which they are decoded. The soft-output of the linear or decision feedback receiver for user i of base j can be written in the common form

$$v_{ij} = \mathbf{f}_{ij}^{\dagger}\left(\mathbf{y}_j - \sum_{k=1}^{i-1}\mathbf{b}_{kij}\sqrt{w_{kj}g_{kjj}}\hat{x}_{kj}\right)\,. \tag{3}$$

$\mathbf{f}_{ij}$ is the feedforward filter. The feedback filters $\mathbf{b}_{kij} = \mathbf{0}$ for linear receivers (no decision feedback) and for decision feedback receivers, we let $\mathbf{b}_{kij} = \mathbf{s}_{kjj}$ for $k = 1, \cdots, i-1$, without loss of generality [1]. It is assumed that $|\mathbf{f}_{ij}^{\dagger}\mathbf{s}_{ijj}|^2 > 0$. The symbol † denotes complex-conjugate transpose. $\hat{x}_{kj}$ denotes the detected (in uncoded transmission) or decoded (in the coded case) symbols of "past" users $1, \cdots, i-1$ relative to user i of base j. There is an issue of decoding delay in coded systems that we will address later in this paper. Implicit in (3) is the assumption that base j has knowledge of the codes, signals, transmit powers and channel gains of the users within its own cell. We do not however assume that it has knowledge of these parameters for out-of-cell users.

A. Single-Cell Multiuser Receivers

For the conventional (or single-user) receiver we have $\mathbf{f}_{ij} = \mathbf{s}_{ijj}$. The conventional decision feedback (C-DF) receiver has the same feedforward filters but with non-zero feedback filters $\mathbf{b}_{kij} = \mathbf{s}_{kjj}$ for $k = 1, \cdots, i-1$.

A simple linear multiuser receiver that can be implemented at each base without knowledge of the system parameters corresponding to out-of-cell users is the single-cell decorrelator which for base j is given as

$$\mathbf{f}_{ij}^{\dagger} = \operatorname{row}_i(\mathbf{S}_{jj}^{\dagger}\mathbf{S}_{jj})^{-1}\mathbf{S}_{jj}^{\dagger}\,. \tag{4}$$

Note that such a receiver tunes-out the interference for each user that arises due to the other users in the same cell but ignores interference from out-of-cell users.

The single-cell decorrelating decision feedback (D-DF) receiver is specified next. The feedback filter coefficients are $\mathbf{b}_{kij} = \mathbf{s}_{kjj}$ for $k = 1, \cdots, i-1$ as before, and the feedforward filters are defined as follows: consider a partition of the columns of $\mathbf{S}_{jj}$ into two matrices, one that contains the first $i-1$ columns and the second that contains the other $K_j - i + 1$ columns.

Accordingly, let $\mathbf{S}_{jj} = \left[\bar{\mathbf{S}}_{jj}(i)\mathbf{S}_{jj}(i)\right]$. The feedforward filter of the single-cell D-DF receiver of user i is given as

$$\mathbf{f}_{ij}^{\dagger} = \mathrm{row}_1 \left(\mathbf{S}_{jj}^{\dagger}(i)\mathbf{S}_{jj}(i)\right)^{-1} \mathbf{S}_{jj}^{\dagger}(i). \tag{5}$$

This receiver subtracts interference contributed by the already decoded users in cell j and decorrelates interference from the as yet undecoded users in cell j [13] without consideration for out-of-cell users.

B. Maximum Signal-to-Interference Ratio Receivers

Ignoring interference from out-of-cell users may be acceptable if path losses are high enough so that the received interference from out of cell users is sufficiently low. When this is not the case, multiuser receivers must take into account the presence of inter-cell interference. In this section we specify the maximum SIR linear and decision feedback receivers for the cellular channel.

Let us define matrices $\{\mathbf{Q}_{jl}\}$ for $j,l \in \{1,2,\cdots,B\}$, with $Q_{jl}(i,k)$ denoting the $(i,k)^{th}$ element of $\mathbf{Q}_{jl}$, as follows: the matrices $\mathbf{Q}_{jj}$ are defined for $j = 1,\cdots,B$ as

$$Q_{jj}(i,k) = \begin{cases} |\mathbf{f}_{ij}^{\dagger}(\mathbf{s}_{kjj} - \mathbf{b}_{kij})|^2 & \text{if } k < i \\ 0 & \text{if } k = i \\ |\mathbf{f}_{ij}^{\dagger}\mathbf{s}_{kjj}|^2 & \text{if } k > i \end{cases} \tag{6}$$

for $i,k \in \{1,\cdots,K_j\}$ and the matrices $\mathbf{Q}_{jl}$ for $j,l \in \{1,2,\cdots,B\}$ with $j \neq l$ are defined as

$$Q_{jl}(i,k) = |\mathbf{f}_{ij}^{\dagger}\mathbf{s}_{klj}^{-}|^2 + |\mathbf{f}_{ij}^{\dagger}\mathbf{s}_{klj}^{+}|^2 \tag{7}$$

for $i \in \{1,\cdots,K_j\}$ and $k \in \{1,\cdots,K_l\}$.

Assuming perfect feedback in the case of decision feedback receivers, the SIR in the soft output v_{ij} of user i in base j can be expressed as

$$\gamma_{ij} = \frac{w_{ij} g_{ijj} |\mathbf{f}_{ij}^{\dagger}\mathbf{s}_{ijj}|^2}{\sum_{l=1}^{B} I_{ijl} + \sigma_j^2 \mathbf{f}_{ij}^{\dagger}\mathbf{f}_{ij}}, \tag{8}$$

where the interference from cell l is given as $I_{ijl} = \sum_{k=1}^{K_l} Q_{jl}(i,k) w_{kj} g_{klj}$.

The maximum signal-to-interference ratio (MSIR) linear receiver (i.e., one that maximizes γ_{ij} in (8) over the filter coefficients $\mathbf{f}_{ij}$) is described by the vectors $\mathbf{f}_{ij}^{msir}$ that satisfy the equations (cf. [7] [8])

$$\mathbf{A}_{ij}^{(L)} \mathbf{f}_{ij}^{\mathrm{msir}} = \mathbf{s}_{ijj} \qquad i = 1,\cdots K_j, \quad j = 1,\cdots,B, \tag{9}$$

where

$$\mathbf{A}_{ij}^{(L)} = \left(\mathbf{S}_{jj}\mathbf{G}_{jj}\mathbf{W}_j(\mathbf{S}_{jj})^{\dagger} + \sum_{l\neq j}^{B} \mathbf{S}_{lj}^{-}\mathbf{G}_{lj}\mathbf{W}_l(\mathbf{S}_{lj}^{-})^{\dagger} + \sum_{l\neq j}^{B} \mathbf{S}_{lj}^{+}\mathbf{G}_{lj}\mathbf{W}_l(\mathbf{S}_{lj}^{+})^{\dagger} + \sigma_j^2\mathbf{I} \right). \tag{10}$$

The maximum signal-to-interference ratio (MSIR) decision feedback receiver is described by the vectors $\mathbf{f}_{ij}^{msir-df}$ that satisfy the equations [8] [1]

$$\mathbf{A}_{ij}^{(DF)} \mathbf{f}_{ij}^{\mathrm{msir-df}} = \mathbf{s}_{ijj} \qquad i = 1,\cdots K_j, \quad j = 1,\cdots,B, \tag{11}$$

where

$$\mathbf{A}_{ij}^{(DF)} = \left(\sum_{k=i}^{K_j} g_{kjj} w_{kj} \mathbf{s}_{kjj}\mathbf{s}_{kjj}^{\dagger} + \sum_{l\neq j}^{B} \mathbf{S}_{lj}^{-}\mathbf{G}_{lj}\mathbf{W}_l(\mathbf{S}_{lj}^{-})^{\dagger} + \sum_{l\neq j}^{B} \mathbf{S}_{lj}^{+}\mathbf{G}_{lj}\mathbf{W}_l(\mathbf{S}_{lj}^{+})^{\dagger} + \sigma_j^2\mathbf{I} \right). \tag{12}$$

The MSIR-DF receiver has the interesting property that it achieves the total capacity in a single-cell channel [1]. It can have a markedly improved performance relative to the MSIR linear receiver in both the single- and multi-cell channels.

Note that in contrast to the single-cell decorrelator with and without feedback, the MSIR linear and MSIR-DF receivers depend on the system parameters corresponding to all users including those outside the cell of interest. Since such parameters are not available at each base station, it is of interest to consider blind adaptive multiuser receivers for the cellular channel. In this regard, the minimum output energy (MOE) [9] and signal subspace tracking [10] approaches, proposed in the context of a single-cell channel, to obtain blind adaptive algorithms that converge to the linear MSIR receiver, can be used for each user at a base station for the multi-cell model of this paper as well. Blind algorithms that are faster than the one based on the MOE criterion of [9] but with similar and even lesser computational complexity have been obtained more recently in [11]. Any of these methods can also be used to obtain algorithms that converge to the blind MSIR-DF receiver. This is most easily deduced from the key point made in [8] and [1] that the MSIR-DF receiver for each user is a MSIR linear receiver designed for, and acting on, the received signal with interference contributions from the already decoded users subtracted, with the filter coefficients determined under the assumption that interference cancellation is perfect.

III. OPTIMUM POWER CONTROL

Given that each base uses an arbitrary but fixed linear or decision feedback receiver, consider the problem of minimizing total transmitted power subject to the constraints that the SIR of each user in the network is not less than some target value:

$$\begin{aligned} &\min \sum_{j=1}^{B}\sum_{i=1}^{K_j} w_{ij} \\ &\text{s.t.} \quad \gamma_{ij} \geq \gamma_{ij}^{*} \quad i = 1,2,\cdots,K_j, \quad j = 1,2,\cdots,B. \end{aligned} \tag{13}$$

For $j \in \{1,2,\cdots,B\}$, define the diagonal matrices $\mathbf{D}_j$ according to

$$\mathbf{D}_j = \operatorname{diag}\left\{\gamma_{1j}^{*}/g_{1jj}|\mathbf{f}_{1j}^{\dagger}\mathbf{s}_{1jj}|^2, \cdots, \gamma_{K_jj}^{*}/g_{K_jjj}|\mathbf{f}_{K_jj}^{\dagger}\mathbf{s}_{K_jjj}|^2\right\}. \tag{14}$$

Let $\mathbf{t}_j = \sigma_j^2\left[\mathbf{f}_{1j}^{\dagger}\mathbf{f}_{1j}, \cdots, \mathbf{f}_{K_jj}^{\dagger}\mathbf{f}_{K_jj}\right]^T$. Define the vector of transmitted powers of users in cell j as $\mathbf{w}_j = [w_{1j}, \cdots, w_{K_jj}]^T$. The SIR constraints in (13) can be written for each j as

$$\mathbf{w}_j \geq \mathbf{D}_j\left[\sum_{l=1}^{B}\mathbf{Q}_{jl}\mathbf{G}_{lj}\mathbf{w}_l + \mathbf{t}_j\right]. \tag{15}$$

The target SIRs are said to be *feasible* if there exist non-negative and finite transmit powers that satisfy the above inequalities. The necessary and sufficient condition for feasibility is stated next.

Define a square, non-negative, block matrix $\mathcal{X}$ with B^2 matrix elements with the $(j,l)^{\text{th}}$ element defined as the matrix product $\mathbf{X}_{jl} = \mathbf{D}_j\mathbf{Q}_{jl}\mathbf{G}_{lj}$ for each $j,l \in \{1,2,\cdots,B\}$. Note that this matrix is not necessarily irreducible (cf. [12]). It is reducible for instance, in the particular case of a single-cell channel when the base employs any decision feedback receiver. There exists a real non-negative eigenvalue $\lambda_0(\mathcal{X})$ of $\mathcal{X}$ with non-negative left and right eigenvectors such that the magnitude of any other eigenvalue of $\mathcal{X}$ is no greater than $\lambda_0(\mathcal{X})$. This eigenvalue is called the maximal eigenvalue of $\mathcal{X}$ [12]. It can be shown that the necessary and sufficient condition for the feasibility of the target SIRs is that $\lambda_0(\mathcal{X}) < 1$. When this condition is satisfied, there is a unique non-negative solution to the problem in (13) which can also be shown to be component-wise minimal. The optimal powers $\{\mathbf{w}_j^*\}$ in fact satisfy the inequalitites in (15) with equality, i.e, they are the solution of the set of equations that result from setting all the SIRs equal to the respective target values. Let $\mathbf{w}$ and $\mathbf{t}$ be vectors obtained by stacking the vectors $\{\mathbf{w}_j\}_{j=1}^B$ and $\{\mathbf{t}_j\}_{j=1}^B$, respectively. Define the matrix $\mathbf{D} = \mathrm{diag}\{\mathbf{d}_1, \cdots, \mathbf{D}_B\}$. Denoting the optimal powers as $\mathbf{w}^*$, we have

$$\mathbf{w}^* = (\mathcal{I} - \mathcal{X})^{-1}\mathbf{D}\mathbf{t}. \tag{16}$$

Implicit in this result is the assumption that feedback is perfect in the case of nonlinear receivers. While not valid in general, it is possible, by decoding the users in the decreasing order of their target SIRs, to mitigate the error propoagation effects to a large extent [13]. In this regard, equal target SIRs may represent something of a worst-case scenario. In such cases, it may be possible to change the problem slightly and set the target SIRs to be slightly different from each other while ensuring that the worst-case target SIR is equal to the required target SIR. The slight increase in the powers required to achieve these higher set of target SIRs may be a small price to offset the effects of error propagation.

The problem of total power minimization under inequalities of the type (15) has been studied before in the literature on power control but without reference to multiuser receivers. Moreover, the results of those studies on feasibility and uniqueness have been found under the assumption that the matrix corresponding to $\mathcal{X}$ is irreducible.

A. Distributed Power Control

The solution in (16) can be computed when all the parameters of the system are known at each base station. In this section, we specify a distributed algorithm that requires much less information.

The optimum powers satisfy the equation $\mathbf{w} = \mathcal{X}\mathbf{w} + \mathbf{D}\mathbf{t}$. There are several iterative methods for solving this systems of linear equations of which the point Jacobi method [14] is particularly simple and relevant to the problem of distributed power control at the mobiles (see also [15]). In particular, consider the distributed algorithm

$$\mathbf{w}(n+1) = \mathcal{X}\mathbf{w}(n) + \mathbf{D}\mathbf{t} \tag{17}$$

which is easily shown to converge to the optimum powers when the target SIRs are feasible. Defining $\epsilon(n) = \mathbf{w}(n) - \mathbf{w}^*$, we see that by subtracting $\mathbf{w}^*$ from both sides of (17), we have $\epsilon(n+1) = \mathcal{X}^{n+1}\epsilon(0)$ so that as $n \to \infty$, $\epsilon(n) \to 0$ since $\mathcal{X}^n \to \mathbf{0}$ as a result of $\lambda_0(\mathcal{X}) < 1$ [12].

Writing (17) element-wise, we have

$$w_{ij}(n+1) = \sum_{l=1}^{B}\sum_{k=1}^{K_l} X_{jl}(i,k) w_{kl}(n) + D_j(i,i) t_{ij} \qquad i = 1, \cdots, K_j, \quad j = 1, \cdots, B, \tag{18}$$

where $X_{jl}(i,k)$ is the $(i,k)^{th}$ element of the matrix $\mathbf{X}_{jl}$ and $D_j(i,i)$ is the i^{th} diagonal element of the matrix $\mathbf{D}_j$ and t_{ij} is the i^{th} element of $\mathbf{t}_j$. The utility of the above distributed algorithm is made more evident by writing it in the equivalent form

$$w_{ij}(n+1) = \gamma_{ij}^* \left(\frac{E[|v_{ij}(n)|^2]}{g_{ijj}|\mathbf{f}_{ij}^\dagger \mathbf{s}_{ijj}|^2} - w_{ij}(n) \right). \tag{19}$$

It is now clear that (19) can be implemented at the $(i,j)^{th}$ mobile or in the j^{th} base station provided it has perfect knowledge of the mean squared value or power of the normalized decision statistic $v_{ij}(n)/\mathbf{f}_{ij}^\dagger \mathbf{s}_{ijj}$, in addition to its target SIR value, the path gain to its own base station and the transmit power in the previous interval.

IV. DISTRIBUTED STOCHASTIC POWER CONTROL

The problem here is to devise a distributed algorithm that iteratively updates the transmit powers of each mobile without perfect knowledge of the power of the decision statistic at the base station corresponding to that mobile. In particular, if the transmit powers are updated once every several symbols, it is possible for the base station to *estimate* the mean squared values of the decision statistics by computing the corresponding sample mean-squared values. In what follows, we specify a synchronous distributed stochastic power control algorithm where the transmit powers are updated (either at the base stations or at the mobiles) iteratively and synchronously once every M symbols. When the power control algorithm is implemented at the base (mobile) stations, the bases communicate the newly updated powers (sample mean-squared values of the decision statistics) to their respective mobiles via low rate feedback channels. The mobiles then transmit the next M symbols at the newly updated powers.

Form M-length blocks of the vectors of matched filter outputs at each base. Let $\mathbf{y}_j(n,m)$ denote the m^{th} vector in the n^{th} block. The model in (1) can be written including the time indices as

$$\begin{aligned}\mathbf{y}_j(n,m) &= \sum_{i=1}^{K_j}\sqrt{w_{ij}(n)g_{ijj}}\mathbf{s}_{ijj}x_{ij}(n,m)+\mathbf{n}_j(n,m)\\ &+\sum_{l\neq j}^{B}\sum_{i=1}^{K_l}\sqrt{w_{il}(n)g_{ilj}}\left(\mathbf{s}_{ilj}^{-}x_{il}^{-}(n,m)+\mathbf{s}_{ilj}^{+}x_{il}^{+}(n,m)\right)\end{aligned} \tag{20}$$

where $w_{il}(n)$ denotes the transmit power of user i of base l during the n^{th} block, and $x(n,m)$ with the appropriate sub- and superscripts denote the corresponding transmitted symbols in the m^{th} duration in the n^{th} block. The decision statistic for user i of cell j is

$$v_{ij}(n,m)=\mathbf{f}_{ij}^{\dagger}\left(\mathbf{y}_j-\sum_{k=1}^{i-1}\mathbf{b}_{kij}\sqrt{w_{kj}(n)g_{kjj}}\hat{x}_{kj}(n,m)\right), \tag{21}$$

where $\hat{x}_{kj}(n,m)$ is the detected symbol in the uncoded case, and it is the decoded and re-encoded symbol in a coded system with decoders that are based on the soft outputs $v_{ij}(n,m)$ obtained during the n^{th} block. The time-averaged power in the decision statistic is given by

$$z_{ij}(n)=\frac{1}{M-1}\sum_{m=1}^{M-1}|v_{ij}(n,m)|^2. \tag{22}$$

Note that in a coded system with decision feedback after decoding and re-encoding (cf. [1]), it is necessary to choose M to be an integer multiple of the block length of the code and to synchronize the times of power updation with the beginning of a new codeword. Consider the following stochastic algorithm, based on the Robbins-Monro stochastic approximation method [16], that iteratively updates the transmit power of mobile i in cell j:

$$w_{ij}(n+1)=w_{ij}(n)+a_n\left(\frac{z_{ij}(n)\gamma_{ij}^{*}}{g_{ijj}|\mathbf{f}_{ij}^{\dagger}\mathbf{s}_{ijj}|^2}-(1+\gamma_{ij}^{*})w_{ij}(n)\right), \tag{23}$$

where $\{a_n\}$ denotes the sequence of step-sizes. The algorithm in (23) is applicable to arbitrary linear and decision feedback multiuser receivers. The special case of the conventional linear receiver reduces to the one proposed in [5]. The chief limitation of the conventional receiver is that, for a given set of target SIRs, feasiblility is assured only for systems with low signal correlations and hence for large spreading gains or bandwidth. Moreover, even when the target SIRs are feasible, the total power expended to achieve these targets is in general much higher compared to that needed for well-designed multiuser receivers, particularly for nonlinear receivers [3].

The mean-square convergence of (23) is stated next. Let $\mathbf{w}_j(n)$ be the vector of power updates of users in cell j in the n^{th} iteration. Let $\mathbf{w}(n)$ be the power vector obtained by stacking $\{\mathbf{w}_j(n)\}_{j=1}^{B}$ and let $\mathbf{w}^*$ be its optimal value.

Theorem: If the target SIRs are feasible, and assuming perfect feedback (in the case of decision-feedback receivers), the decentralized stochastic algorithm in (23) converges to the optimal powers in the mean square sense, i.e.,

$$\lim_{n\to\infty} E\left[||\mathbf{w}(n)-\mathbf{w}^*||^2\right]=0 \tag{24}$$

for a positive-valued step-size sequence a_n that satisfies $\sum_{n=1}^{\infty} a_n=\infty$ and $\sum_{n=1}^{\infty} a_n^2<\infty$.

V. JOINTLY OPTIMAL POWER CONTROL AND MULTIUSER RECEIVERS

Rather than obtaining optimal power control strategies for fixed multiuser receivers, we consider in this section the problem of minimizing the minimum total power required to achieve the QoS objectives over the feedforward and feedback filter coefficients. This approach yields the most power efficient solution when the receivers are constrained to be of the linear or decision feedback type.

A. The Single-Cell Case

For the single-cell model, we seek the jointly optimum pair of a multiuser receiver and its optimum power allocation that requires the least total power to achieve the given target SIRs. Assuming that only users in base 1 are active, we consider the problem:

$$\min_{\substack{\mathbf{f}_{i1}\in C^{N_1}\\ i=1,\cdots K_1}} \quad \min_{\substack{\mathbf{w}_1\geq 0\\ \gamma_{i1}\geq\gamma_{i1}^*}} \quad \sum_{i=1}^{K_1} w_{i1} \quad i=1,2,\cdots,K_1 \tag{25}$$

A similar problem was first posed by the author in [6] and [17] where the solution to the problem of finding the most power efficient linear receiver was obtained for the asymptotic effective energy (rather than the SIR) measure. We adapt that solution to the SIR measure and also consider the more general problem of optimizing over the class of linear and nonlinear receivers. Starting with the C-DF receiver, consider the sequence of receiver and power allocation pairs generated according to the following two-step iterative algorithm which alternately computes the best power vector and the corresponding MSIR-DF receiver:

$$\mathbf{w}_1(n) = (\mathbf{I}-\mathbf{X}_{11}(n-1))^{-1}\mathbf{D}_1(n-1)\mathbf{t}_1(n-1) \tag{26}$$

$$\mathbf{f}_{i1}(n) = \left(\sum_{k=i}^{K_1} g_{k11}w_{k1}(n)\mathbf{s}_{k11}\mathbf{s}_{k11}^{\dagger}+\sigma_1^2\mathbf{I}\right)^{-1}\mathbf{s}_{i11} \qquad i=1,2,\cdots,K_1, \tag{27}$$

where $\mathbf{X}_{11}(n))^{-1},\mathbf{D}_1(n)$ and $\mathbf{t}_1(n)$ are $\mathbf{X}_{11},\mathbf{D}_1$ and $\mathbf{t}_1$ evaluated at the filters $\mathbf{f}_{i1}(n)$ obtained in the n^{th} iteration.

It can be shown that the limit of the above sequence converges to the jointly optimal receiver and power allocation strategy. Of course, one can constrain the receivers to be linear, start with the decorrelator (when it exists), carry out the two-step iteration described above with the linear MSIR receiver in place of the decision feedback MSIR receiver, and arrive at the jointly optimal pair of linear multiuser receiver and its optimum power allocation [17]. However, the total power required in this case is always greater than that required for the decision feedback case.

B. Distributed Power Control: The Multi-Cell Case

In this section, we consider the multi-cell model. Independently of [6] and [17], this problem was also considered in [4] for the case of a symbol synchronous cellular channel and with a restriction on the multiuser receivers to be linear. We consider the more general problem which is to find, among all sets of linear and decision feedback receivers employed at the different base stations and the corresponding power allocation for all users in the network, the most power efficient combination, i.e., the jointly optimal pair of receivers and the transmit powers which requires the least power to achieve the target SIRs of all users in the cellular network, i.e.,

$$\min_{\substack{\mathbf{f}_{ij} \in C^{N_j} \\ i=1,\cdots,K_j \\ j=1,\cdots,B}} \quad \min_{\substack{\mathbf{w} \geq 0 \\ \gamma_{ij} \geq \gamma_{ij}^{*}}} \quad \sum_{j=1}^{B} \sum_{i=1}^{K_j} w_{ij} \quad \begin{matrix} \\ \\ i=1,2,\cdots,K_j \\ j=1,\cdots,B\,. \end{matrix} \tag{28}$$

It can shown that, assuming that the target SIRs are feasible, the following two-step iterative algorithm converges to the optimum pair of receivers and power allocations for all users in the network. Denoting the power allocations of users in base j at time step n as $\mathbf{w}_j(n)$ and the vector obtained by stacking these vectors in order as $\mathbf{w}(n)$, the iterative algorithm at base j is

$$\mathbf{w}_j(n) = \sum_{l=1}^{B} \mathbf{X}_{jl}(n-1)\mathbf{w}_l(n-1) + \mathbf{D}_j(n-1)\mathbf{t}_j(n-1), \tag{29}$$

$$\mathbf{f}_{ij}^{\text{msir-df}}(n) = (\mathbf{A}_{ij}^{(DF)}(\mathbf{w}(n)))^{-1}\mathbf{s}_{ijj} \qquad i=1,\cdots,K_j, \tag{30}$$

where, letting $\mathbf{F}_j(n)$ be the $N_j \times K_j$ matrix whose i^{th} row is $\mathbf{f}_{ij}(n)$ for $i=1,\cdots,K_j$, we indicate the dependence of the quantities $\mathbf{X}_{jl}$, $\mathbf{D}_j$ and $\mathbf{t}_j$ on $\mathbf{F}_j(n)$ by adding the iteration step n in parantheses. Moreover, the matrix $\mathbf{A}_{ij}^{(DF)}(\mathbf{w}(n))$ is the MSIR-DF receiver as defined in (12) with the transmit powers $\mathbf{w}$ replaced by $\mathbf{w}(n)$. One could start with any receiver such as the C-DF receiver to initialize the above algorithm. Note however, that the computations of the iterations of (29) and (30) require a knowledge of all system parameters.

The relevance to distributed implementation of the above algorithm is made more evident by writing it in the equivalent form

$$\mathbf{w}_j(n) = \mathbf{D}_j\mathbf{p}_j(n-1) - \Gamma_j\mathbf{w}_j(n-1) \tag{31}$$

$$\mathbf{f}_{ij}^{\text{msir-df}}(n) = \left(E[\mathbf{y}_j(n)\mathbf{y}_j^{\dagger}(n)] - \sum_{k=1}^{i-1} w_{kj}(n) g_{kjj}\mathbf{s}_{kjj}\mathbf{s}_{kjj}^{\dagger}\right)^{-1} \mathbf{s}_{ijj} \quad i=1,\cdots,K_j, \tag{32}$$

where $\mathbf{p}_j(n) = [p_{1j}(n),\cdots,p_{K_j j}(n)]^T$ is a K_j-length vector of mean-squared values of the decision statistics, i.e., $p_{ij}(n) = E[|v_{ij}(n)|^2]$. Note that all the system parameters involved in the above iteration such as signature signals, path gains, transmit powers and target SIRs for a fixed j involve only the users in cell j. As a result, this iterative solution can be computed at each base station independently of the other base stations provided that base j has perfect knowledge of $E[\mathbf{y}_j(n)\mathbf{y}_j^{\dagger}(n)]$ and the vector of mean squared values of the decision statistics $\mathbf{p}_j(n)$. The problem of how these algorithms must be modified to deal with a lack of such knowledge is dealt with in the next section.

C. Stochastic Power Control for the Blind MSIR-DF Receiver

In this section, we consider the problem of obtaining a stochastic counterpart of the algorithm given in (31) and (32) where we do not assume perfect knowledge of either $E[\mathbf{y}_j(n)\mathbf{y}_j^{\dagger}(n)]$ or $\mathbf{p}_j(n)$.

We consider M-length blocks of the observation vectors as in (20) and divide this block into two parts of length M_1 and M_2 respectively, i.e., $M = M_1 + M_2$. In the first part of the n^{th} block, we adapt to the MSIR-DF receiver, i.e., for $m = 0, 1, \cdots, M_1 - 1$. Using equation (32) and applying the Robbins-Monro stochastic approximation method, we obtain the adaptation rule

$$\mathbf{f}_{ij}(n,m+1) = \mathbf{f}_{ij}(n,m) - a_{n,m+1}\left[\left(\mathbf{y}_j(n,m)\mathbf{y}_j^{\dagger}(n,m) - \sum_{k=1}^{i-1} w_{kj}(n) g_{kjj}\mathbf{s}_{kjj}\mathbf{s}_{kjj}^{\dagger}\right)\mathbf{f}_{ij}(n,m) - \mathbf{s}_{ijj}\right]. \tag{33}$$

The initialization for the above iteration is $\mathbf{f}_{ij}(n,0) = \mathbf{f}_{ij}(n-1, M_1)$. The final filter obtained after M_1 iterations of (33) in the n^{th} block is fixed for the rest of the block, i.e., $\mathbf{f}_{ij}(n,m) = \mathbf{f}_{ij}(n, M_1)$ for $m \in \{M_1 + 1, \cdots, M\}$. The adaptation rule in (33) can be more efficiently implemented by making use of the quantities evaluated in computing the decision statistic for the m^{th} symbol of the n^{th} block. In particular, let

$$v_{ij}(n,m) = r_{ij}(n,m) - \sum_{k=1}^{i-1} B_{kij}(n,m)\sqrt{w_{kj}(n) g_{kjj}\hat{x}_{kj}(n,m)}, \tag{34}$$

where $r_{ij}(n,m) = \mathbf{f}_{ij}^{\dagger}(n,m)\mathbf{y}_j(n,m)$ is the output of the feedforward filter and $B_{kij}(n,m) = \mathbf{f}_{ij}^{\dagger}(n,m)\mathbf{s}_{kjj}$ are the feedback coefficients for user i of cell j for $k = 1, \cdots, i-1$. In terms of these quantities, the algorithm in (33) can be implemented as

$$\mathbf{f}_{ij}(n,m+1) = \mathbf{f}_{ij}(n,m) - a_{n,m+1}\left[\mathbf{y}_j(n,m) r_{ij}^{*}(n,m) - \sum_{k=1}^{i-1} w_{kj}(n) g_{kjj} B_{kij}^{*}(n,m)\mathbf{s}_{kjj} - \mathbf{s}_{ijj}\right]. \tag{35}$$

In the second part of the n^{th} block where the receive filter coefficients remain unchanged, we compute the sample mean squared value of the decision statistic in (34) according to

$$z_{ij}(n) = \frac{1}{M_2}\sum_{m=M_1}^{M-1} |v_{ij}(n,m)|^2, \tag{36}$$

and update the transmit power to be used in the $(n+1)^{th}$ block according to the rule

$$w_{ij}(n+1) = (1 - b_n) w_{ij}(n) + b_n \gamma_{ij}^{*}\left(\frac{z_{ij}(n)}{g_{ijj}|\mathbf{f}_{ij}^{\dagger}(n, M_1)\mathbf{s}_{ijj}|^2} - w_{ij}(n)\right), \tag{37}$$

where b_n is a time-varying step-size sequence that satisfies the conditions $\sum_{n=1}^{\infty} b_n = \infty$ and $\sum_{n=1}^{\infty} b_n^2 < \infty$. The choice of the step-size sequences $a_{n,m}$ and the constant b in the sequence $b_n = b/n$ for the receiver and power adaptation in (35) and (37), and the effect of choice of the integers M_1 and M_2 are subjects for further investigation.

VI. NUMERICAL SIMULATIONS

We consider a 6-cell system with base stations located on a rectangular grid with (x, y) coordinates $(.5 + i, .5 + j)$ (in Km) for $i = 0, 1, 2$ and $j = 0, 1$. There are 10 users per cell and their (x, y) coordinates are chosen randomly according to a uniform distribution over a circle of radius .5 Km with the base as center. The path loss exponent is $\alpha = 4$. The block size is $M = 20$ and $\sigma_j^2 = 10^{-14}$ W. The signature sequences of the sixty users are chosen to be random binary sequences of length 20 each. The signals of all users in the network are assumed to arrive in symbol synchronism at each base.

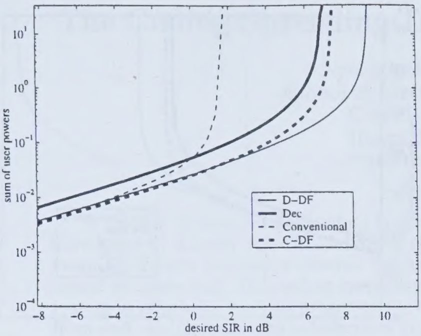


Fig. 1. Minimum total power vesus common target SIR for the matched filter receiver and the single-cell decorrelator (Dec), the C-DF and the single-cell D-DF.

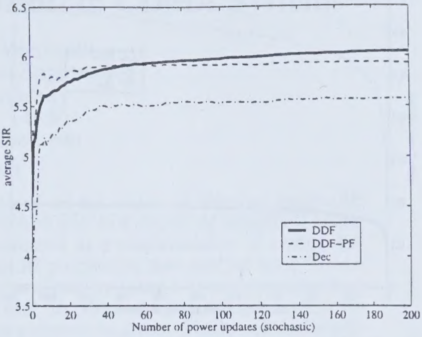


Fig. 2. Average SIR trajectory of the 60 users with the stochastic power control algorithm in (23).

Figure 1 shows the minimum total power required as a function of the common target SIR for the conventional (single-user) receiver, the single-cell decorrelator, the C-DF and the single-cell D-DF receivers. Note that for each of the receivers, there is a feasibility interval of common achievable target SIRs. the upper limit of The feasibility interval for the matched filter is significantly smaller than that for the multiuser receivers. Among the latter, the nonlinear C-DF and D-DF receivers have larger feasibility intervals than the single-cell decorrelator. Note that at a common target SIR of 6 dB, the total power required by the single-cell decorrelator is nearly 10 times that required by the D-DF receiver with the power savings being even higher at larger target SIRs. In implementing the stochastic power control algorithms, we set the common target SIR to be 6 dB.

Next, we present simulation results for the stochastic power control algorithm of equation (23) implemented for each of the six bases with a common target SIR of 6dB. The initial powers at each base were chosen to be locally optimal in that they are equal to the optimum powers under the incorrect assumption that there are no transmissions in interfering cells. Such powers can be computed without knowing the signal and gain parameters of users outside the cell of interest.

Figure 2 shows the average SIR for the single-cell decorrelator and the single-cell D-DF, as a function of the iteration index. The single-cell D-DF with perfect feedback is also shown to assess the effects of error propagation. The value of the constant a in the step-size $a_n = a/n$ was chosen to be best (experimentally) for each of the three cases. The single-cell decorrelator is much further (relative to the D-DF) from converging in 200 iterations because 6dB is much closer to its highest achievable target SIR than it is for the single-cell D-DF.

Figures 3 depicts the total power expended for each of the three cases considered in Figure 2 as a function of the iteration index. The D-DF with and without feedback are close to the minimum total power requirement in very few iterations since they achieve the target SIRs quickly. On the other hand, the total power for the decorrelator in 200 iterations is a little over half of what it will eventually be with increasing n.

Next, we plot in Figure 4, the minimum total power versus the common target SIR for the jointly optimal power allocation and MSIR-DF receiver. The fixed receivers considered in Figure 1 are also included for the sake of comparison. Note that with the MSIR-DF receiver target SIRs of up to 18.4 dB are feasible compared to the 9.2 dB for the single-cell D-DF receiver. Furthermore, at a common target SIR of say 9 dB, the MSIR-DF receiver requires a total power that is 11 times less than what is needed by the single-cell D-DF.

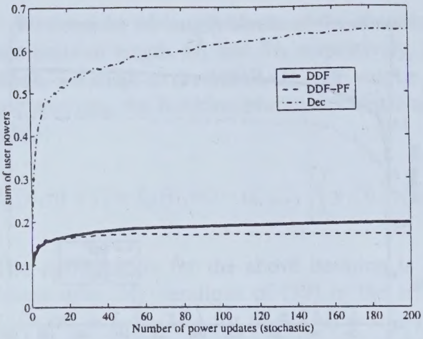


Fig. 3. Minimum total power vesus common target SIR for the matched filter receiver and the single-cell decorrelator (Dec), the C-DF and the single-cell D-DF.

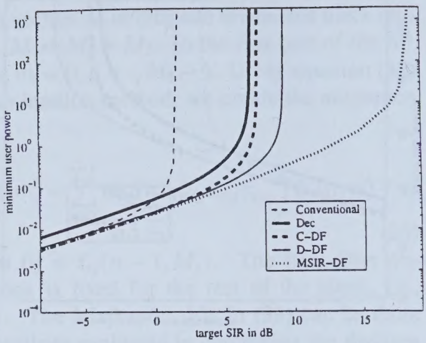


Fig. 4. Minimum total power versus common target SIR for the MSIR-DF receiver compared with the fixed receivers of Figure 1.

VII. CONCLUSION

A proper integration of multiuser receivers at the physical layer and power control algorithms at the network layer can have a significant impact on the design of bandwidth- and power-efficient cellular networks.

Acknowledgement

The author is grateful to Deepak Das for generating the numerical simulations of this paper.

REFERENCES

[1] M. K. Varanasi and T. Guess, "Optimum decision feedback multiuser equalization with successive decoding achieves the total capacity of the gaussian multiple-access channel," *Proc. 31st Annual Asilomar Conf. on Signals, Systems, and Computers*, pp. 1405–1409, Nov. 1996.

[2] P. S. Kumar and J. Holtzman, "Power control for a spread spectrum system with multiuser receivers," *Proc IEEE Personal Indoor and Mobile Radio Communications, PIMRC 1995*, pp. 955–959, Sept. 1995.

[3] M. K. Varanasi, "Power control for multiuser detection," *Proc. Conf. Inform. Sc. Sys.*, pp. 866–873, Mar. 1996.

[4] S. Ulukus and R. Yates, "Adaptive power control with SIR maximizing multiuser detectors," *Proc. Conf. Inform. Sc. Sys.*, pp. 707–712, Mar. 1997.

[5] S. Ulukus and R. Yates, "Stochastic power control for cellular radio systems," *IEEE Trans. Commun.*, vol. 46, pp. 784–798, June 1998.

[6] M. K. Varanasi, "Power control for multiuser detectors in CDMA systems," *Office of Naval Research Proposal*, vol. CU Proposal No. 1096.03.1364B, Oct. 1996.

[7] U. Madhow and M. L. Honig, "MMSE interference suppression for direct-sequence spread-spectrum CDMA," *IEEE Trans. Commun.*, vol. 42, pp. 3178–3188, Dec. 1994.

[8] T. Guess and M. K. Varanasi, "Multiuser decision feedback receivers for the general gaussian multiple-access channel," in *Proc. 34st Annual Allerton Conf. on Commun., Contr. and Comput.*, (Monticello, IL), pp. 190–199, Oct. 1996.

[9] M. L. Honig, U. Madhow, and S. Verdu, "Blind adaptive multiuser detection," *IEEE Trans. Inform. Theory*, vol. 41, pp. 994–960, July 1995.

[10] X. Wang and H. V. Poor, "Blind multiuser detection: a subspace approach," *IEEE Trans. Inform. Theory*, vol. 44, pp. 677–690, Mar. 1998.

[11] D. Das and M. K. Varanasi, "Blind adaptive multiuser detection for cellular systems," *submitted, IEEE Trans. Commun.*, Oct. 1999.

[12] E. Seneta, *Non-negative matrices*. New York: John Wiley, 1973.

[13] M. K. Varanasi, "Decision feedback multiuser detection: A systematic approach," *IEEE Trans. Inform. Theory*, vol. 45, pp. 219–240, Jan. 1999.

[14] L. A. Hageman and D. M. Young, *Applied Iterative Methods*. London: Academic Press, 1971.

[15] G. Foschini and Z. Miljanic, "A simple distributed autonomous power control algorithm and its convergence," *IEEE Trans. Veh. Tech.*, vol. 42, pp. 641–646, Nov. 1993.

[16] H. Robbins and S. Monro, "A stochastic approximation method," *Annals of Math. Stat.*, vol. 22, pp. 400–407, 1951.

[17] M. K. Varanasi, "Multiuser receivers with power control," *National Science Foundation Proposal*, vol. CU Proposal No. 0497.05.0526B, Apr. 1997.

The Coding-Spreading Tradeoff in CDMA Systems[1]

Venugopal V. Veeravalli
School of Electrical Engineering
Cornell University
Ithaca, NY 14850
venu@ee.cornell.edu

Abstract

A general definition of spreading is given based on the notion of Shannon bandwidth introduced by Massey [1]. This definition is shown to lead to a important separation result: every bandwidth expansion scheme can be expressed as a concatenation of coding followed by spreading. The coding-spreading tradeoff problem is then studied for a CDMA system in which the receiver processes the receiver signal by using a linear user-separating front-end, which feeds into autonomous single-user decoders. An example illustrating the coding-spreading tradeoff optimization for such a system is given. Single-cell and multi-cell scenarios are considered in the optimization, and a comparision is made of the user capacities and spectral efficiencies that can be achieved with conventional matched filter and linear MMSE front-ends.

I Fourier and Shannon Bandwidths

Consider the space of functions that are time-limited to the interval $[0, T]$ and approximately bandlimited to a baseband bandwidth of W, i.e., most of the energy is contained in the range of frequencies $[-W, W]$. We denote this space by $\mathcal{S}(T; W)$, and we note that $\mathcal{S}(T; W)$ dimension D_F given by (see e.g., [2, p. 294]):

$$D_F \approx 2WT\,.$$

We refer to D_F as the *Fourier dimension* of the signal space. The Prolate Spheroidal Wave Functions (PSWF's) form a basis for this space, but if $T \gg 1/W$ and we ignore edge effects, we have the following simpler set of orthonormal basis functions that are formed by translations of sinc pulses.

$$\phi_i(t) = \sqrt{2W}\operatorname{sinc}(2Wt - (i - 0.5)), \;\; i = 1, 2, \ldots, D_F\,. \tag{1}$$

Consider the following single user communications problem. Suppose our goal is to transmit one of M possible symbols (equivalently a sequence of $\log_2 M$ bits) in time $[0, T]$, using one of M possible signals from the set $\{s_1(t), s_2(t), \ldots s_M(t), t \in [0, T]\}$. We do not restrict the form of signals in any way except that the signal set occupies a Fourier bandwidth of W, i.e., $[-W, W]$ is the smallest range of frequencies that encompasses the essential bandwidths of all the signals in the set. Note that W is also the bandwidth of the random process that represents the signal that is transmitted on the channel (one that equals $\{s_m(t)\}$ with prior probability p_m, $m = 1, 2, \ldots, M$).

It is clear that each $\{s_m(t)\}$ belongs to $\mathcal{S}(T; W)$, and can hence be represented by a D_F-dim vector $s_m = [s_{m,1}\; s_{m,2}\; \cdots\; s_{m,D_F}]^\top$, with

$$s_m(t) = \sum_{i=1}^{D_F} s_{m,i}\; \phi_i(t)\,. \tag{2}$$

Without loss of generality, we may assume that the $D_F \le M$; since if $D_F > M$, the M signals can be assigned orthogonal directions with the remaining dimensions being unused.

We now note that the signal set may actually span a space that has dimension that is smaller than D_F. This leads to the following definition of Shannon bandwidth that was introduced by Massey in [1].

---

[1] This research is supported in part by an NSF CAREER/PECASE grant under CCR-9733204.

***Definition* 1.** The dimension of Span$\{s_1, \ldots, s_M\}$ is called the *Shannon dimension* of the signal set, and is denoted by D_S. The ratio $D_S/2T$, which represents half the number of dimensions per second occupied by the signal set, is called the *Shannon bandwidth* of the signal set, and is denoted by B. Clearly $D_S \leq D_F$ and $B \leq W$. □

The notion of Shannon bandwidth was used by Massey to precisely define a spread-spectrum system as one for which $B < W$ (actually $B \ll W$). The ratio $\gamma = W/B$ is defined to be the spreading factor. Based on this definition, Massey gives several examples in [1] that illustrate the difference between spread-spectrum and non-spread-spectrum systems. In the following section, we use this notion of Shannon bandwidth to define the operation of *spreading*, and more importantly to distinguish it from another way to introduce bandwidth redundancy, namely, *coding*.

II Spreading as a Linear Mapping in Signal Space

Because standard direct sequence spreading can be considered as equivalent to repitition coding, it is not immediatedly clear how to distinguish the the operations of spreading and coding. Based on Massey's definition of a spread-spectrum system, it is easy to see that spreading cannot increase capacity in AWGN. Thus one way to distinguish the two operations is to recognize that coding improves performance (provides positive coding gain) on AWGN channels, whereas spreading leaves performance unchanged (provides zero coding gain).

It is tempting to define spreading as any bandwidth redundancy scheme that increases the Fourier bandwidth, while preserving the Shannon bandwidth. However, such a definition may not be consistent with the notion that spreading provides zero coding gain in AWGN, as the following example illustrates.

***Example* 1.** Consider a signal set, with $D_S = D_F = 2$, consisting of the signals

$$s_1 = [\tfrac{1}{\sqrt{2}}\ \tfrac{1}{\sqrt{2}}]^\top, \text{ and } s_2 = [1\ 0]^\top .$$

Consider the bandwidth redundancy scheme that increases D_F to 3 in such a way that the new signal set is

$$s_1' = [0\ 1\ 0], \text{ and } s_2' = [1\ 0\ 0] .$$

Clearly D_S remains unchanged, but we achieve coding gain in AWGN, since the mapping increases the Euclidean distance between the signals while preserving their energies. □

To rule out such special cases and still be consistent with Massey's definition of a spread spectrum system, we define spreading in the following fashion as a normalized linear mapping[2].

***Definition* 2.** On time interval $[0, T]$, a bandwidth redundancy scheme that increases the Fourier bandwidth from W to $W' > W$ is called *spreading* if it can be expressed as a linear mapping $\boldsymbol{L}$ from $\mathcal{S}(T; W)$ to $\mathcal{S}(T; W')$ that satisfies $\boldsymbol{L}^\top \boldsymbol{L} = \boldsymbol{I}$. □

The above definition of spreading generalizes the two standard ways of spreading: direct sequence spreading and frequency hopping.

***Example* 2.** *Direct Sequence Spreading.* Consider a M-ary signal set on $\mathcal{S}(T; W)$. Using the basis functions of (1), we have $s_m = [s_{m,1}\ s_{m,2}\ \cdots\ s_{m,D_F}]^\top,\ m = 1, 2, \ldots, M$.

---

[2] It is easy to check that the mapping in Example 1 that takes the set $\{s_1, s_2\}$ to the set $\{s_1', s_2'\}$ is not linear.

Direct sequence spreading involves replacing $\phi_i(t)$ in (2) by the chip signal $c_i(t)$ that is given by:

$$c_i(t) = \sum_{n=1}^{N} c_{i,n}\ \psi_{(i-1)N+n}(t)$$

where

$$\psi_j(t) = \sqrt{2WN}\mathrm{sinc}(2WNt - (j-0.5))\,,\ \ j = 1, \ldots, ND_F$$

are basis functions for $\mathcal{S}(T, NW)$. Thus the spread signal is given by

$$s'_m(t) = \sum_{i=1}^{D_F} s_{m,i}\ c_i(t) = \sum_{i=1}^{D_F}\sum_{n=1}^{N} c_{i,n} s_{m,i}\ \psi_{(i-1)N+n}(t)\,.$$

Of course, since $\{s'_m(t)\}$ belongs to $\mathcal{S}(T, NW)$, we can write

$$s'_m(t) = \sum_{j=1}^{ND_F} s'_{m,j}\psi_j(t)\,.$$

Now, if we define $\boldsymbol{s}'_m = [s'_{m,1}\ s_{m,2}\ \cdots\ s'_{m,ND_F}]^\top$ and $\boldsymbol{c}_i = [c_{i,1}\ c_{i,2}\ \cdots\ c_{i,N}]^\top$, then

$$\boldsymbol{s}'_m = \boldsymbol{L}\boldsymbol{s}_m\,,\ \ m = 1, 2, \ldots, M$$

where

$$\boldsymbol{L} = \begin{bmatrix} \boldsymbol{c}_1 & 0 & \cdots & 0 \\ 0 & \boldsymbol{c}_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & \boldsymbol{c}_{D_F} \end{bmatrix}$$

If we normalize the spreading sequences so that $\boldsymbol{c}_i^\top\boldsymbol{c}_i = 1$, it is easy to see that $\boldsymbol{L}^\top\boldsymbol{L} = \boldsymbol{I}$. □

Similarly, if we define our basis functions appropriately, it can be shown that frequency hopping is a linear mapping in signal space with

$$\boldsymbol{L} = \begin{bmatrix} \boldsymbol{e}_{j(1)} & 0 & \cdots & 0 \\ 0 & \boldsymbol{e}_{j(2)} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & \boldsymbol{e}_{j(D_F)} \end{bmatrix}$$

where $\boldsymbol{e}_i$ denotes the unit N-dim vector with a 1 in the i-th position, and $j(i) \in \{1, 2, \ldots, N\}$. Generalized spreading, as defined in Definition 2, clearly preserves the Shannon bandwidth while increasing the Fourier bandwidth. Furthermore, we have the following result, which follows readily from the result that spreading preserves distances between the signals in the signal set.

***Proposition* 1.** Generalized spreading leaves performance unchanged (provides zero coding gain) on AWGN channels. □

III Separation of Coding and Spreading

To complete our description of the dichotomy between coding and spreading, we define the operation of coding in terms of Shannon and Fourier bandwidths.

***Definition* 3.** On the time interval $[0, T]$, consider a scheme that increases the Fourier bandwidth of an M-ary signal set from W to $W' > W$, and the Shannon bandwidth from B to $B' > B$. This scheme is called *coding* if it improves performance on AWGN channels, and if

$$W'/W = B'/B\,. \tag{3}$$

Note that condition (3) ensures that the coding operation does not "waste" bandwidth. Also note that coding is necessarily a non-linear mapping from $\mathcal{S}(T;W)$ to $\mathcal{S}(T;W')$.

Definitions 2 and 3 lead naturally to the following separation result.

***Proposition* 2.** *Separation of Coding and Spreading.* Consider an M-ary signal set that belongs to $\mathcal{S}(T;W)$, and has Shannon bandwidth $B = W$. Consider any bandwidth expansion scheme that increases the Fourier bandwidth of the signal set to $W' > W$, and the Shannon bandwidth to B'. (Clearly, $W \le B' \le W'$.) This scheme can be written as a concatenation of coding from $\mathcal{S}(T;W)$ to $\mathcal{S}(T;B')$, followed by spreading from $\mathcal{S}(T;B')$ to $\mathcal{S}(T;W')$.

Proof. Let $D_F = 2WT$, $D'_S = 2B'T$ and $D'_F = 2W'T$. The M-ary signal set can be represented by the D_F-dim vectors $\{s_1, s_2, \ldots, s_M\}$ and the D'_F-dim vectors $\{s'_1, s'_2, \ldots, s'_M\}$ in $\mathcal{S}(T;W)$ and $\mathcal{S}(T;W')$ respectively. The bandwidth expansion scheme may be described by a mapping f that satisfies:

$$s'_m = f(s_m)\,, \quad \text{for } m = 1, 2, \ldots, M\,.$$

If $B' > B$, this mapping is necessarily nonlinear.

Note that the signals $\{s'_m\}$ are D'_F-dim vectors that span a space of dimension D'_S. Thus, we may identify a set of D'_S orthonormal vectors $\{\boldsymbol{l}_j\}$, and write

$$s'_m = \sum_{j=1}^{D'_S} \tilde{s}_{j,m} \boldsymbol{l}_j\,. \tag{4}$$

If we set $\boldsymbol{L} = [\boldsymbol{l}_1 \ \cdots \ \boldsymbol{l}_{D'_S}]$, then it is clear that $\boldsymbol{L}^\top \boldsymbol{L} = \boldsymbol{I}$.

Associated with each s'_m is a D'_S-dim vector $\tilde{s}_m = [\tilde{s}_{1,m} \ \cdots \tilde{s}_{D'_S,m}]^\top$. Using (4) and the fact that $\boldsymbol{L}^\top \boldsymbol{L} = \boldsymbol{I}$, we can write

$$\tilde{s}_m = \boldsymbol{L}^\top s'_m = \boldsymbol{L}^\top f(s_m) =: g(s_m)\,, \quad \text{for } m = 1, 2, \ldots, M.$$

Now, g is a mapping from $\mathcal{S}(T;W)$ to $\mathcal{S}(T, B')$ that increases the Fourier and Shannon bandwidths by same amounts. Thus g is the *coding* part of the bandwidth expansion scheme. The *spreading* part is obviously described by the mapping $\boldsymbol{L}$, since

$$s'_m = \boldsymbol{L}\tilde{s}_m\,, \quad \text{for } m = 1, 2, \ldots, M.$$

with $\boldsymbol{L}^\top \boldsymbol{L} = \boldsymbol{I}$. □

IV The Coding-Spreading Tradeoff Problem

Given any bandwidth expansion scheme, Proposition 2 allows us to identify and separate the coding and spreading components of this scheme. It is clear that coding and spreading contribute differently to system performance. But how should a fixed bandwidth expansion factor be optimally allocated between coding and spreading?

For illustration, consider the tradeoff problem for a single-user, AWGN communication system. Since spreading cannot increase channel capacity, the capacity maximizing solution puts all of the bandwidth expansion into coding. However, as noted in [1], spreading need not reduce capacity too much, and if practical constraints such as decoding complexity are taken into accout, using a significant fraction of the bandwidth expansion for spreading may be justified.

Now consider the coding-spreading tradeoff in the context of Code Division Multiple Access (CDMA) systems. For an AWGN channel with equal-rate, equal-power users, the capacity maximizing solution again favors all coding [3]. Of course this solution assumes an optimum joint decoding scheme at the receiver. For synchronous users with spreading factor N equal to number of users K, it is possible to orthogonalize users so as to not incur any loss in capacity. Even when $K > N$ it is possible to find spreading sequences that incur no loss in capacity [3].

The tradeoff problem is much more interesting in practical CDMA systems, where complexity constraints favor adopting a user-separating front-end at the receiver that is followed by autonomous single-user decoders. We restrict our attention to such CDMA systems in the remainder of this paper.

Consider the symmetric situation where each one of K users independently needs to send one of M symbols ($\log_2 M$ information bits) in interval $[0, T]$. Before bandwidth expansion, the m-th component of the signal set of the k-th user is denoted by $\{s_{k,m}(t)\}$. We assume that all of these signal sets have Fourier bandwidth W that is equal to the Shannon bandwidth B. For example, each user might use Binary Phase Shift Keying (BPSK) with sinc bit pulses to represent the information bits.

Suppose all K signals are coded by a factor B'/W and then spread by factor W'/W, with $W < B' < W'$. For user k, let g_k denote the nonlinear mapping (code-book) that defines the coding, and let $\boldsymbol{L}_k$ denote the linear mapping that defines the spreading. Then the coding and spreading operations are, respectively, described by:

$$\tilde{s}_{k,m} = g_k\left(s_{k,m}\right) \ , \quad \text{and } s'_{k,m} = \boldsymbol{L}_k \tilde{s}_{k,m} \, . \tag{5}$$

The mappings $\{g_k\}$ and $\{\boldsymbol{L}_k\}$ could be different across the users, although in practical CDMA systems, such as the one based on the IS-95 [4] standard, the code-books are identical.

The received signal in AWGN is given by

$$\boldsymbol{r} = \sum_{k=1}^{K} s'_k + \boldsymbol{w} \tag{6}$$

where $\boldsymbol{w} \sim \mathcal{N}(0, N_0 \boldsymbol{I})$.

The optimum receiver would form estimates of the information symbols of all users jointly, using a decoder that exploits information about both the spreading and coding components of all users' signals. As mentioned previously, we will assume a suboptimum but practical receiver that processes the received signal in two stages. The front-end produces K outputs $\{\boldsymbol{v}_k\}_{k=1}^{K}$, with $\boldsymbol{v}_k$ being the "best" estimate of the encoder output $\tilde{s}_k$ of user k. The estimates $\{\boldsymbol{v}_k\}$ are then fed to autonomous single-user decoders.

A reasonable criterion for producing the estimates $\{\boldsymbol{v}_k\}$ from $\boldsymbol{r}$ is the Minimum Mean-Squared-Error (MMSE) criterion (also see [5]), where the goal is to pick the $\{\boldsymbol{v}_k\}$ to minimize the mean-squared-error (MSE) given by

$$\mathrm{MSE} = \mathrm{E}\left[\sum_{k=1}^{K} \|v_k - \tilde{s}_k\|^2\right] . \tag{7}$$

The expectation in (7) is over the distribution of the noise and the prior distribution on the symbols of the users.

Note that the probability distribution of the code-vector $\{\tilde{s}_k\}$ is a function of the prior distribution on the information symbols and the code-book $\{g_k\}$. Since the users send independent pieces of information, the code-vectors $\{\tilde{s}_k\}$ are mutually independent across users. However, the individual components of each code-vector $\tilde{s}_k$ are necessarily dependent (for a non-trivial code). Nevertheless, we make the following assumption.

***Assumption* 1.** The code-vectors $\{\tilde{s}_k\}$ are mutually independent. Furthermore, the components of each $\tilde{s}_k$ are zero mean and uncorrelated, i.e.,

$$\mathrm{E}[\tilde{s}_k] = 0 \ , \text{ and } \mathrm{E}\left[\tilde{s}_k \tilde{s}_k^{\top}\right] = \frac{\mathcal{E}_k}{D'_S} I$$

where $\mathcal{E}_k$ is the average energy of $\tilde{s}_k$. □

While Assumption 1 appears to be restrictive, it holds for many coding schemes of interest. For example, consider constructing the code-book g_k from a (D'_S, D_F) error control code $\mathcal{C}_k$ that is linear over a finite field, say GF(q) (this assumes of course that $M = q^{D_F}$). Let c_k denote a generic codeword of $\mathcal{C}_k$. If we consider a particular position, say the i-th, it is easy to show from linearity that $c_{k,i}$ takes on each value in GF(q) the same number of times if we look across code-book $\mathcal{C}_k$. To construct g_k from $\mathcal{C}_k$, we simply map the elements of GF(q) to $\Re$. We note that to maximize distance under given average power constraints, the mapping to $\Re$ must be symmetric around the origin. With the assumption that the information symbols are equally likely, it easily follows that Assumption 1 holds.

The front-end stage that produces the estimates $\{v_k\}$ could in general use knowledge of both the code-books $\{g_k\}$ and the spreading matrices $\{L_k\}$ to minimize the MSE. However, if the front-end is constrained to be linear to reduce complexity, we have the following result, whose proof follows from the standard derivation of the MMSE receiver [6, Chapter 6].

***Proposition* 3.** Under Assumption 1, the linear MMSE solution for the receiver front-end depends only on the spreading matrices of the various users, and not their code-books. □

Propostion 3 illustrates the separation of coding and spreading at the receiver. Coding does not help with linear signal separation at the front-end, and spreading of course does not aid the single-user decoders in combating AWGN. However, it is important to note that a nonlinear front-end can make use of knowledge of the codebooks to improve performance. An example of such a non-linear front-end is an interference cancellation scheme that uses the code-books of the interferers to reconstruct their signals for cancellation.

In the following section we give a simple example of how we may optimize the coding-spreading tradeoff for a CDMA system with a linear user-separating front-end.

V Example

Consider a CDMA system in which the users are constrained to send binary signals. Each user needs to send $\mu = \log_2 M$ bits of information in some long time interval T. Without bandwidth expansion, the minimum bandwidth required is $W = \mu/2T$. For such a binary signal set, $D_F = \mu$ and $B = W$. Consider a bandwidth expansion scheme that expands the bandwidth to $W' = \nu W$, coding at rate r, followed by spreading by a factor N. Then

$$\nu = \frac{N}{r} .$$

Spreading is achieved using a direct sequence approach, where each code bit (symbol) is spread by a unit energy chip waveform as in Example 2. *Random* spreading is assumed, i.e., the spreading sequences are independent from code bit to code bit and across users. We assume a synchronous system, although similar results will be obtained for asynchronous systems as long as bandwidth restrictions are correctly imposed on the chip waveforms [7]. The sum of the signals from K *equal power* users is received in AWGN with spectral height of $N_0/2$.

Let $\mathcal{E}_b$, $\mathcal{E}_s$ and $\mathcal{E}_c$, denote, respectively, the energies per information bit, code bit, and chip. Also define the corresponding SNR's by $\gamma_b = \mathcal{E}_b/N_0$, $\gamma_s = \mathcal{E}_s/N_0$ and $\gamma_c = \mathcal{E}_c/N_0$. Then

$$\gamma_s = N\gamma_c \,, \text{ and } \gamma_b = \nu\gamma_c \,.$$

After projection on to basis function of the $2W'T$-dim signal space (i.e., chip-matched filtering) and normalization, we get the following model for the received vector for one code bit interval:

$$\boldsymbol{r} = \gamma_s \sum_{k=1}^{K} b_k \boldsymbol{c}_k + \boldsymbol{w}$$

where $\boldsymbol{w} \sim \mathcal{N}(0, \boldsymbol{I})$, the code bits $b_k \in \{-1, +1\}$, and the $\boldsymbol{c}_k$'s are binary $\{-\frac{1}{\sqrt{N}}, +\frac{1}{\sqrt{N}}\}$ vectors with $\mathbf{c}_k^\top \boldsymbol{c}_k = 1$.

Sufficient statistics for decision making based on $\boldsymbol{r}$ may be obtained by projecting on the vectors $\{\boldsymbol{c}_k\}$ to form:

$$\boldsymbol{y} = \boldsymbol{C}^\top \boldsymbol{r} = \sqrt{2\gamma_s}\boldsymbol{R}\boldsymbol{b} + \tilde{\boldsymbol{w}}$$

where the columns of $\boldsymbol{C}$ are the spreading vectors $\{\boldsymbol{c}_k\}$, the matrix $\boldsymbol{R} = \boldsymbol{C}^\top \boldsymbol{C}$, and $\tilde{\boldsymbol{w}} \sim \mathcal{N}(0, \boldsymbol{R})$.

For a linear front-end, the code bit estimates are based on the vector

$$\boldsymbol{z} = \boldsymbol{H}\boldsymbol{y} \,.$$

We restrict our attention to two linear front-ends, the Matched Filter (MF) for which $\boldsymbol{H} = \boldsymbol{I}$, and the optimum linear MMSE front-end for which $\boldsymbol{H} = [\boldsymbol{I} + 2\gamma_s \boldsymbol{R}]^{-1}$ (see, e.g. [6]).

For *hard* decision-making, $\text{sgn}(z_k)$ is sent to the decoder for user k. For *soft* decision-making z_k (or some appropriate function of z_k) is sent to the decoder.

The goal in the coding-spreading tradeoff optimization is to pick the coding and spreading factors to maximize performance. Of course the spreading factor N determines the coding rate, so we only need to vary N. Since we are constraining the information rates of the users (to $2W$ bits/s or one bit every ν chips), a reasonable measure of performance is the largest number of users K_ν that can transmit their bits reliably on the channel. The value K_ν is a function of N, and the optimum spreading factor is given by

$$N^\star = \arg\max_N K_\nu(N) \,.$$

If we denote the maximum value by $K_\nu^\star$, the spectral efficiency in total number of bits that can be sent reliably on the channel is given by $K_\nu^\star/\nu$. Note that the maximum spectral efficiency can be obtained more easily if we allow ν to vary with K and N. Using this approach, the asymptotic spectral efficiency for large K and N is obtained for various scenarios without binary constraints on the input by Verdu and Shamai [3].

For a practical system, the code would be chosen from a family of multiple rate codes, and $K_\nu(N)$ would be the largest number of users accomodated with a bit error rate of less than some threshold (say 10^{-3}). But we may draw useful conclusions about the coding-spreading tradeoff more easily by assuming ideal coding. In this case $K_\nu(N)$ is computed as follows. For fixed K and N, we compute the capacity $\mathcal{C}(K,N)$ of the effective single-user channel (between the output of the encoder and that of the receiver front-end) for any user, say user 1. Since the code rate must be less than $\mathcal{C}(K,N)$ for reliable transmission, we have

$$K_\nu(N) = \text{maximum value of } K \text{ such that } \frac{N}{\mathcal{C}(K,N)} < \nu\,.$$

For hard decision decoding, the effective single-user channel is a binary symmetric channel (BSC) that is time-invariant and memoryless by the assumption of random spreading. Thus

$$\mathcal{C}(K,N) = 1 - H(\bar{\mathrm{P}}_e(K,N))$$

where H is the binary entropy function, and $\bar{\mathrm{P}}_e$ is the average bit error probability over all the code bits. For the MF and MMSE front-ends, $\bar{\mathrm{P}}_e$ is given by:

$$\begin{aligned}\bar{\mathrm{P}}_e &= \mathrm{E}_{\boldsymbol{R}}\, \frac{1}{2^K} \sum_{b\in\{-1,+1\}^K} Q\left(\sqrt{\frac{2\gamma_s}{(\boldsymbol{HRH})_{1,1}}}\left[(\boldsymbol{HR})_{1,1} + \sum_{j=2}^{K}(\boldsymbol{HR})_{1,j}b_j\right]\right)\\ &= \mathrm{E}_{\boldsymbol{R}}\, Q\left(\sqrt{\frac{2\gamma_s}{(\boldsymbol{HRH})_{1,1}}}\left[(\boldsymbol{HR})_{1,1} + \sum_{j=2}^{K}(\boldsymbol{HR})_{1,j}\right]\right)\end{aligned} \tag{8}$$

where $\boldsymbol{H} = \boldsymbol{I}$ and $\boldsymbol{H} = [\boldsymbol{I} + 2\gamma_s\boldsymbol{R}]^{-1}$ for the MF and MMSE front-ends, respectively. The second line in (8) follows because, with random sequences, we may set the bits of all the interferers to $+1$ without affecting the expression for the average.

For soft decision decoding, the single-user channel is a binary input, continuous output channel. If we approximate the conditional pdf of the output by a Gaussian, then [8, p. 273]

$$\mathcal{C}(K,N) = -\frac{1}{2}\log_2 2\pi e \;-\; \int_{-\infty}^{\infty} p(y)\log_2 p(y)\,dy$$

where $p(y)$ is mixture of $\mathcal{N}(\alpha,1)$ and $\mathcal{N}(-\alpha,1)$ with equal weights, with α being a measure of the SIR at the channel output. For the MF and MMSE front-ends, it is easy to show that:

$$\alpha = \mathrm{E}_{\boldsymbol{R}}\left[\frac{\sqrt{2\gamma_s}(\boldsymbol{HR})_{1,1}}{\sqrt{(\boldsymbol{HRH})_{1,1} + 2\gamma_s\sum_{j=2}^{K}[(\boldsymbol{HR})_{1,j}]^2}}\right] \tag{9}$$

where $\boldsymbol{H} = \boldsymbol{I}$ and $\boldsymbol{H} = [\boldsymbol{I} + 2\gamma_s\boldsymbol{R}]^{-1}$ for the MF and MMSE front-ends, respectively.

A Numerical Results

We first consider the "single-cell" wireless communication system, where all the users in an isolated cell are received with equal power (i.e., they are perfectly power controlled) at the base station and the only interference is from thermal AWGN. Plots (a) and (b) of Fig. 1 show $K_\nu(N)$ for this scenario for the hard- and soft-decision decoding, respectively. As expected,

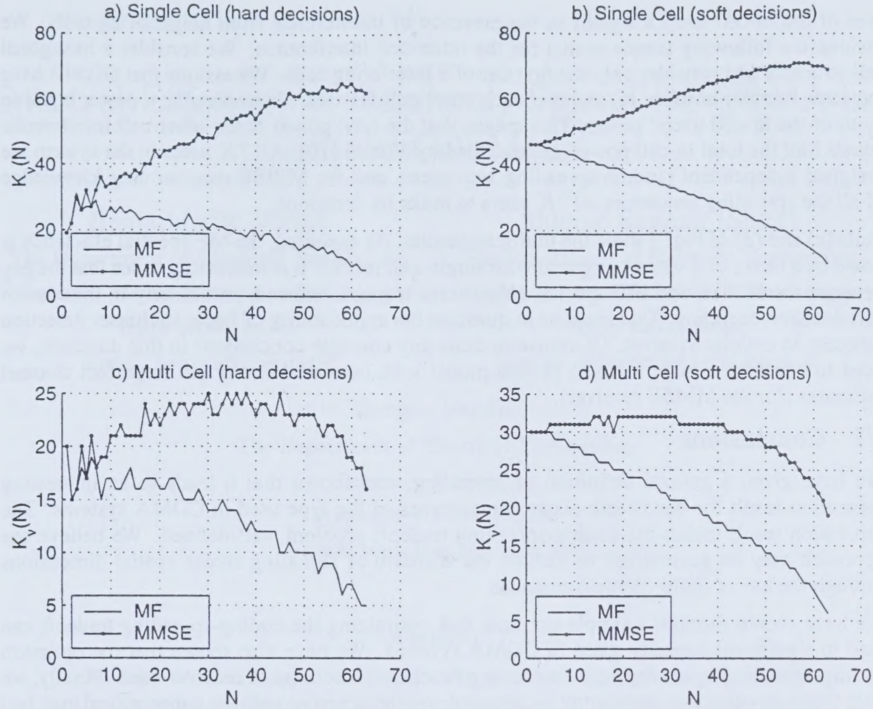


Figure 1: User capacity per cell as a function of the spreading factor for various scenarios. In all cases the bandwidth expansion factor $\nu = 64$, and the information bit SNR $\gamma_b = 18$ dB. The results were obtained by averaging over 2000 independent random spreading sequence sets.

small values of N are favored for the MF front-end, while values of N close to ν are favored for the MMSE front-end. We can also see that coding gives diminishing returns for the MF receiver; a major portion of the bandwidth expansion can be given to spreading without decreasing the user capacity significantly.

For hard decisions, the spectral efficiency of the MF, $\eta^{\text{MF}} = K_\nu^\star/\nu$ equals $30/64 \approx 0.47$ bits/chip, and matches well with the large N asymptotic value of $\frac{\log_2 e}{\pi}$ predicted by Hui [9]. For soft decisions, η^{MF} equals $46/64 \approx 0.72$ bits/chip, and again matches well the value of $\frac{\log_2 e}{2}$ given in [9].

For the MMSE front-end, η^{MMSE} is around 1 bit/chip for hard decisions, and is slightly larger for soft decisions. Asymptotic values for η^{MMSE} are derived in [3] without binary constraints on the input, and hence these results cannot be used for comparison. However, it is straightforward to extend the analysis of [3] to predict the values of η^{MMSE} observed in Fig. 1. It is interesting to note that the MMSE spectral efficiency with random sequences can be higher than the maximum efficiency of 1 bit/chip achievable with orthogonal sequences and $K = N$. This is in contrast to the case without binary constraints where it is known that orthogonalizing users with $K = N$ does not incur any loss in capacity (see, e.g. [3]).

We also give results for a multicell wireless system, in which the base station receives the sum of the in-cell users's signals in the presence of interference from neighboring cells. We assume the following simple model for the other-cell interference. We consider a hexagonal cell structure and consider only the first tier of 6 interfering cells. We assume that all cells have the same number of users K, and that each other-cell interferer is received at a power equal to $\frac{1}{12}$-th of the in-cell users' power. This means that the total power in the other-cell interference equals half the total in-cell power, as described by Viterbi [10]. All $7K$ users in the system are assigned independent random spreading sequences, and the MMSE receiver uses knowledge of all the spreading sequences all $7K$ users to make its decisions.

Plots (c) and (d) of Fig. 1 show the multicell results. As expected, the MF spectral efficiency is down by a factor of $\frac{2}{3}$ when compared with single-cell results. It is interesting to see that the gap between the MMSE and MF spectral efficiencies is much reduced, particularly in the case of soft decision decoding. This leads us to question the applicability of linear multiuser detection schemes in cellular systems. Of course to draw any concrete conclusions in this direction, we need to consider a more realistic system model with non-ideal codes and imperfect channel estimates (for the MMSE receiver).

VI Conclusions

We have given a general definition of spreading, and shown that it leads to an interesting separation result for bandwidth expansion schemes of the type used in CDMA systems. The separation result makes the coding-spreading tradeoff problem well-defined. We believe our approach may be generalized to include the scenario of signaling across spatial dimensions through the use of multi-element antennas.

We have shown through a simple example that optimizing the coding-spreading tradeoff can lead to significant capacity gains in CDMA systems. We have also shown that the optimum coding-spreading operating point is a strong function of the type of receiver used. Finally, we note that many analyses comparing multiuser detection schemes with the conventional matched filter receiver have either ignored coding or used the same code rate for both receivers. We have shown that for a correct comparison of two alternative receiver structures for CDMA, it is important to consider each of them at their optimum coding-spreading operating points.

Acknowledgements. The author is grateful to Ashok Mantravadi and Mehul Motani for helpful discussions and for computing the numerical results presented in this paper.

References

[1] J. L. Massey, "Information theory aspects of spread-spectrum communications," in *Proc. IEEE ISSSTA'94*, (Oulu, Finland), pp. 16–20, July 1994.

[2] J. Wozencraft and I. Jacobs, *Principle of Communication Engineering*. New York: John Wiley and Sons, 1965.

[3] S. Verdu and S. Shamai, "Spectral efficiency of CDMA with random spreading," *IEEE Trans. Inform. Th.*, vol. 45, pp. 622–640, Mar. 1999.

[4] TIA/EIA/IS-95, *Mobile Station-Base Station Compatibility Standard for Dual-Mode Wideband Spread Spectrum Cellular System*. Telecommunications Industry Association, July 1993.

[5] M. Rupf, F. Tarkoy, and J. L. Massey, "User-separating demodulation for code-division multiple-access systems," *IEEE J. Sel. Areas Commun.*, vol. 12, pp. 786–95, June 1994.

[6] S. Verdu, *Multiuser Detection*. Cambridge, UK: Cambridge University Press, 1998.

[7] A. Mantravadi and V. V. Veeravalli, "On discrete sufficient statistics for asynchronous band-limited CDMA systems," *IEEE Trans. Commun.*, Dec. 1998. Submitted.

[8] R. Blahut, *Principles and Practice of Information Theory*. Reading, MA: Addison-Wesley, 1987.

[9] J. Y. N. Hui, "Throughput analysis for code division multiple accessing of the spread spectrum channel," *IEEE J. Sel. Areas Commun.*, vol. 2, pp. 482–486, July 1984.

[10] A. J. Viterbi, "Error-correcting coding for CDMA systems," in *Proc. IEEE ISSSTA '94*, (Oulu, Finland), pp. 22–26, July 1994.

Spectral Efficiency of Randomly Spread DS-CDMA in a Multi-Cell Model

Benjamin M. Zaidel
The Department of Electrical Engineering,
Technion - Israel Institute of Technology,
Haifa 32000, Israel.
e-mail: bennyz@inter.net.il

Shlomo Shamai (Shitz)
The Department of Electrical Engineering,
Technion - Israel Institute of Technology,
Haifa 32000, Israel.
e-mail: sshlomo@ee.technion.ac.il

Sergio Verdú
The Department of Electrical Engineering,
Princeton University,
Princeton, NJ 08544 USA.
e-mail: verdu@ee.princeton.edu

Abstract

A simple multi-cell uplink communication model is suggested and analyzed for optimally coded randomly spread direct sequence code division multiple access (DS-CDMA) multiuser detection strategies. We adhere to Wyner's (1994) infinite linear cell array model, according to which only adjacent cell interference is present, and characterized by a *single* parameter $0 \leq \alpha \leq 1$. We confine our discussion to asymptotic analysis where both the number of users and the processing gain go to infinity, while their ratio is kept fixed and finite. We focus on single cell-site processing and consider the matched filter detector, "optimum" detection with adjacent cell interference treated as Gaussian noise, the linear MMSE detector, and a Decision-Feedback MMSE detector with linear MMSE processing of adjacent cell interference. Spectral efficiency is evaluated under three power control strategies: equal received powers (for all users), equals rates, and a maximal spectral efficiency policy. Comparative results demonstrate how performance of the different detectors is affected by the introduction of inter-cell interference, and what is the penalty associated with the randomly spread coded DS-CDMA strategy.

1 Introduction

Information theoretic analysis of direct sequence code division multiple access (DS-CDMA) systems is of great interest in recent years, due to the rapid development of commercial cellular systems employing this multi-accessing strategy. Results and observations of major importance, for a *single cell* DS-CDMA system, were recently published in [1], and [2] (see also references therein). These works explicitly relate to CDMA systems with *random* spreading sequences, and the limiting scenario is examined, where both the number of users and the processing gain go to infinity, while their ratio is kept fixed and *finite*. Assuming equal powers, expressions are presented in [1] for spectral efficiencies of the optimum detector, the matched filter detector, the decorrelator, and the linear minimum mean squared error (MMSE) detector. Signal to interference plus noise ratios at the output of linear detectors (the matched filter detector, the linear MMSE detector, and the decorrelating detector) are presented in [2], and an extension of these results to fading channels with multi-antenna reception can be found in [3].

Our work addresses *multi-cell* cellular systems using the attractive cellular model suggested by Wyner in [4]. This simple model allows for analytical tractability on the one hand, while giving insight to "real life" systems on the other. Accordingly, the system's cells compose an infinite *linear* array, where the received signal at each cell site is the sum of the signals received from intra-cell users, plus a factor α $(0 \leq \alpha \leq 1)$ times the sum of the signals of users in the two adjacent cells[1], as received at their cell sites. This signal is accompanied by an ambient Gaussian noise. The multi-cell effect on performance is thus specified by a *single* parameter (α).

We assume, as in [4], non-fading channels, and analyze the spectral efficiency obtained by employing *optimally* coded *randomly* spread DS-CDMA multi-user detection. As in [1] and [2], we consider the limiting case in which, denoting by K the number of users (assumed constant and equal in all cells), and by N the spreading factor (processing gain), $K, N \to \infty$, while $\frac{K}{N} = \beta$ remains constant. Assuming *single cell-site processing*, four types of multiuser detection strategies are considered. We first consider the *"conventional" matched filter detector* which treats *all* interference (either intra-cell or inter-cell) as additive white Gaussian noise. The second detector "optimally" detects the transmissions of *intra-cell* users, while treating *inter-cell* interference as additive white Gaussian noise. This detector shall be henceforth referred to as *the single-cell "optimum" detector* (SCO). Next we consider the *linear MMSE detector* which makes use of knowing the signature sequences of interfering users (both intra-cell and in adjacent cells) to suppress their interference by means of a linear MMSE filter. The last detector to be considered is a *decision feedback (DF) MMSE detector* with decision-feedback detection performed over transmissions of intra-cell users, while inter-cell interference is suppressed by means of a linear MMSE filter. We emphasize that neither the linear MMSE detector, nor the DF-MMSE detector, try to *decode* the transmissions of users from adjacent cells (which might be impossible if α is small). In fact, the cell site detector may actually be ignorant regarding codebooks or code-mask sequences employed in other cells, but is aware, as usually is the case in practice, of the signature sequences of all users in adjacent cells. In addition to the above it is also assumed that all detectors are provided with the required knowledge regarding the received powers of the interfering signals[2].

The above detectors are examined under three power control strategies. First we consider the equal powers strategy, which assigns equal *received* powers to all users. Next, we consider the equal rates strategy which employs a power assignment function such that all users attain equal rates (in the sense of information theoretic capacity). The last strategy we discuss is the strategy which attains the maximum spectral efficiency.

This paper is organized as follows. In section 2 we present the system model, and the equations through which the spectral efficiency of the four detectors is obtained. The following sections 3, 4, and 5, discuss the three power control strategies mentioned above. Section 6 ends the paper with a summary and some concluding remarks.

---

[1] No interference form cells further away is assumed.

[2] The use of adaptive MMSE detection may be attractive with this respect, as it makes no distinction between intra-cell and adjacent-cell interference.

2 System Model and Multiuser Detection Strategies

2.1 Multi-Cell System Model

Following [4] and [5], we consider the uplink of a fully synchronous cellular CDMA system whose cells are ordered in an *infinite* linear array. Using the standard discrete time equivalent channel representation, the signal vector received at an arbitrary cell site, at the discrete time related to the transmission of the ith symbol, is given by

$$\boldsymbol{y}_i = \boldsymbol{S}_i \boldsymbol{x}_i + \alpha \boldsymbol{S}_i^- \boldsymbol{x}_i^- + \alpha \boldsymbol{S}_i^+ \boldsymbol{x}_i^+ + \boldsymbol{n}_i \ . \tag{2-1}$$

The vector $\boldsymbol{x}_i = [x_{1,i}, \ldots, x_{K,i}]^T$ in (2-1) comprises the K code symbols received from intra-cell users at the ith discrete time. The vectors $\boldsymbol{x}_i^\pm = [x_{1,i}^\pm, \ldots, x_{K,i}^\pm]^T$ denote the vectors of symbols received at adjacent cell sites from the users operating in these adjacent cells. These symbols are assumed to be i.i.d., Gaussian[3], with $E\{x_{k,j}\} = 0$ and $E\{x_{k,j}^2\} = P_k \ \forall\, k, j$, where P_k is the power controlled received power of the kth user. The matrices $\boldsymbol{S}_i$ and $\boldsymbol{S}_i^\pm$ are $N \times K$ matrices, whose columns are the N-chip long spreading (signature) sequences of the K users in the considered cell and in its adjacent cells, respectively. According to our assumptions the entries of the above matrices are treated as i.i.d. zero mean random variables, with variance $1/N$. The vector $\boldsymbol{n}_i$ represents a zero mean white Gaussian noise vector, with $E\{\boldsymbol{n}_i \boldsymbol{n}_i^\dagger\} = \sigma^2 \boldsymbol{I}$, $\forall i$.

We analyze the effect of power control strategies on the overall system performance. By "power control strategy" we mean a system defined centrally controlled function, which assigns a *received*[4] power level to each of the intra-cell users. According to our model exactly the same power control strategy is applied to all cells. Denoting the normalized power (SNR) of the kth user by $\tilde{P}_k = \frac{P_k}{\sigma^2}$, it is assumed that as the number of users grows to infinity, the discrete power control function $\{\tilde{P}_k\}_{k=1}^K$ goes to a limit given by a function of a continuous argument, denoted henceforth by $f(x)$ $(0 < x \leq 1)$[5], and sums can thus be replaced by integrals. The above power control function is assumed to be subject to an average power constraint, which is expressed in the continuous form by

$$\int_0^1 f(x)\, dx = \bar{P} \ . \tag{2-2}$$

The fundamental figure of merit for system performance is the per cell *spectral efficiency* (see [1]), defined as the total number of bits per chip that can be transmitted arbitrarily reliably in a single cell. Denoting the signal to interference plus noise ratio (SIR) function at the the output of a detector by $g(x)$ (according to our limiting continuous argument assumption), and following central limit results which have shown that the interference at the output of each of the detectors is well approximated by a Gaussian noise[6], the spectral efficiency is given by (see next subsection for applicability to the SCO detector)

$$\tilde{C} = \frac{\beta}{2} \int_0^1 \log(1 + g(x))\, dx \ . \tag{2-3}$$

---

[3] This is justified by assuming that the codebooks of all users are chosen randomly, governed by an underlying i.i.d. Gaussian distribution per symbol, and *independently* for each message transmission (see [5]).

[4] It is assumed that the users adjust their transmission power so as to be received at the prescribed level.

[5] That is, $\tilde{P}_k \xrightarrow[K \to \infty]{} f(\frac{k}{K})$, $k = 1, 2, \ldots, K$, with $x = \frac{k}{K}$.

[6] See [1], [6], and [7] for justification of this Gaussian approximation.

In the following subsection we generally describe how the spectral efficiency of all four multiuser detectors is obtained in terms of the power control function. We use the notation of $(\cdot)_{\mathrm{mf}}$, $(\cdot)_{\mathrm{sco}}$, $(\cdot)_{\mathrm{ms}}$, and $(\cdot)_{\mathrm{df}}$ to designate entries related to the matched filter detector, the SCO detector, the linear MMSE detector, and the DF-MMSE detector, respectively.

2.2 Spectral Efficiency of the Multiuser Detectors

The matched filter detector simply passes the received signal through a filter matched to the signature sequence of the user of interest, thus treating all interfering signals as a pure additive white Gaussian noise. It is hence straightforward to show that the spectral efficiency of the matched filter detector is given by

$$\tilde{C}_{\mathrm{mf}} = \frac{\beta}{2}\int_0^1 \log\left(1 + \frac{f(x)}{1+\beta\,(1+2\alpha^2)\,\bar{P}}\right) dx \ . \tag{2-4}$$

The spectral efficiency of the SCO detector is most conveniently obtained using the well known observation (e.g. see [8], and [1]) that the DF-MMSE detector achieves the optimum spectral efficiency, assuming Gaussian input symbols. Following [2], the SIR function at the output of the above equivalent DF-MMSE detector is determined by solving the following implicit equation

$$g_{\mathrm{sco}}(x) = \frac{f(x)}{1+2\beta\alpha^2\bar{P}+\beta\int_x^1 \frac{f(x)f(\zeta)}{f(x)+f(\zeta)g_{\mathrm{sco}}(x)}\,d\zeta} \ . \tag{2-5}$$

The spectral efficiency of the SCO detector can be evaluated through (2-3).

Similarly, the SIRs at the output of the linear MMSE detector and the DF-MMSE detector are obtained, respectively, via

$$g_{\mathrm{ms}}(x) = \frac{f(x)}{1+\beta\int_0^1\left[\frac{f(x)f(\zeta)}{f(x)+f(\zeta)g_{\mathrm{ms}}(x)} + \frac{2\alpha^2 f(x)f(\zeta)}{f(x)+\alpha^2 f(\zeta)g_{\mathrm{ms}}(x)}\right]d\zeta} \ , \tag{2-6}$$

and

$$g_{\mathrm{df}}(x) = \frac{f(x)}{1+\beta\left[\int_x^1 \frac{f(x)f(\zeta)}{f(x)+f(\zeta)g_{\mathrm{df}}(x)}\,d\zeta + \int_0^1 \frac{2\alpha^2 f(x)f(\zeta)}{f(x)+\alpha^2 f(\zeta)g_{\mathrm{df}}(x)}\,d\zeta\right]} \ . \tag{2-7}$$

Again we use (2-3) to evaluate the spectral efficiency of both detectors.

We note that when different systems (with possibly different spreading gains and data rates) are to be compared, it is imperative to express the spectral efficiency in terms of $\frac{E_b}{N_0}$, which is done through the relation $\bar{P} = \frac{2}{\beta}\,\tilde{C}\,\frac{E_b}{N_0}$ (see [1]). However, for simplicity of notations we express equations in terms the normalized power. For details of the derivation of the above equations see [9].

3 Equal Powers

Satisfying the average power constraint of (2-2), the equal powers strategy simply implies setting $f(x) = \bar{P}$, $\forall\, 0 < x \leq 1$. Substituting the above power control function in (2-4), the spectral efficiency of the matched filter detector is given by

$$\tilde{C}_{\mathrm{mf}} = \frac{\beta}{2}\log\left[1+\frac{\bar{P}}{1+\beta\,(1+2\alpha^2)\,\bar{P}}\right] \ . \tag{3-1}$$

As can be seen, the above result reaches a limit as the average power (or $\frac{E_b}{N_0}$) grows without bound, which is in accordance with the interference limited behavior of the matched filter detector. The same behavior is observed as we take $\beta \to \infty$, which is also optimum in terms of spectral efficiency.

Denoting $\bar{P}_{\text{eq}} \triangleq \frac{\bar{P}}{1+2\beta\alpha^2\bar{P}}$, and substituting $f(x) = \bar{P}$ into (2-5), it can be shown that the spectral efficiency of the SCO detector is given by

$$\tilde{C}_{\text{sco}} = \frac{\beta}{2}\log\left[1+\bar{P}_{\text{eq}} - \frac{1}{4}\mathcal{F}(\bar{P}_{\text{eq}},\beta)\right] + \frac{1}{2}\log\left[1+\bar{P}_{\text{eq}}\beta - \frac{1}{4}\mathcal{F}(\bar{P}_{\text{eq}},\beta)\right] - \frac{\log e}{8\bar{P}_{\text{eq}}}\mathcal{F}(\bar{P}_{\text{eq}},\beta)\,, \tag{3-2}$$

where $\mathcal{F}(x,z) \triangleq \left(\sqrt{x\left(1+\sqrt{z}\right)^2+1} - \sqrt{x\left(1-\sqrt{z}\right)^2+1}\,\right)^2$. As with the matched filter detector, it is observed that the above spectral efficiency reaches a limit as we let $\frac{E_b}{N_0} \to \infty$ (note that $\bar{P}_{\text{eq}} \to \frac{1}{2\beta\alpha^2}$ as $\frac{E_b}{N_0} \to \infty$). The interference limited behavior of the SCO detector emanates from the fact that inter-cell interference is treated as additive white Gaussian noise. Again, $\beta \to \infty$ is optimum in terms of spectral efficiency.

The SIR at the output of the linear MMSE detector is given by the positive root of the cubic equation obtained by substituting $f(x) = \bar{P}$ into (2-6). As can be seen the resulting SIR function is a constant, and thus the spectral efficiency of the linear MMSE detector is simply given by $\tilde{C}_{\text{ms}} = \frac{\beta}{2}\log(1+g_{\text{ms}})$. In contrast to the two previous detectors, the linear MMSE filter is not interference limited if β is appropriately chosen. It can be shown that the spectral efficiency of the linear MMSE detector grows without bound as we take $\frac{E_b}{N_0} \to \infty$, as long as we keep $\beta \leq \frac{1}{3}$ (optimum choice is $\beta < \frac{1}{3}$ for large $\frac{E_b}{N_0}$). Taking $\beta \to \infty$, the spectral efficiency of the linear MMSE detector coincides with that of the matched filter, as expected from the single-cell results of [1].

Substituting $f(x) = \bar{P}$ into (2-7) results again in a cubic equation, whose positive root determines the value of the SIR function $g_{\text{df}}(x)$, $\forall\, x \in [0,1]$ (note that here the SIR function is not a constant). Obtaining the SIR function, the resulting spectral efficiency can be evaluated through (2-3). As with the linear MMSE detector, the spectral efficiency of the DF-MMSE detector can be shown to grow without bound as $\frac{E_b}{N_0} \to \infty$, if β is appropriately chosen (again the optimum choice is $\beta < \frac{1}{3}$ for large $\frac{E_b}{N_0}$). Taking $\beta \to \infty$ it is observed that the spectral efficiency of the DF-MMSE detector coincides with that of the SCO detector, which is in agreement with the behavior of the linear MMSE detector for $\beta \to \infty$.

Fig. 1 compares the spectral efficiencies of the four detectors discussed above with $\alpha = \frac{1}{2}$ and optimum choice of β. As can be seen the spectral efficiencies of the linear MMSE and the DF-MMSE detectors, coincide with those of the matched filter detector and the SCO detector, for low $\frac{E_b}{N_0}$ where it is optimum to choose $\beta \to \infty$. However as $\frac{E_b}{N_0}$ increases, beyond some critical values of $\frac{E_b}{N_0}$ the optimum choice of β for the two MMSE based detectors decreases, eventually becoming lower than $\frac{1}{3}$, and the spectral efficiency of these detectors grows without bound with $\frac{E_b}{N_0}$. This explains the "knee effect" observed in the spectral efficiency curves (we note that the "knee effect" is not observed when the spectral efficiency is plotted for a *fixed* value of β). The slope of the spectral efficiency of the DF-MMSE detector is however steeper than that of the linear MMSE detector. One can also notice that beyond some critical value of $\frac{E_b}{N_0}$ (around $16dB$ in Fig. 1) the spectral efficiency of the linear MMSE detector surpasses that of the SCO detector.

Another interesting issue is the penalty due to random spreading. Following [5], one can observe that the spectral efficiencies of the matched filter detector and the SCO

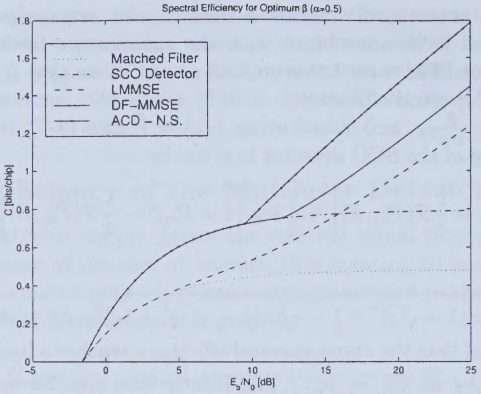


Figure 1: Spectral efficiency comparison with equal powers for $\alpha = \frac{1}{2}$, and an optimum choice of β ("N.S." denotes - No Spreading).

detector, coincide with those of the two corresponding detectors without the constraint of random spreading, when we take $\beta \to \infty$ (which is the optimum choice, c.f. Fig. 3). Another interesting comparison is with respect to a detector which knows the *codebooks* of users in adjacent cells, and either decodes their transmissions as well, or treats them as additive Gaussian noise (referred to as ACD - "adjacent-cell decoder"). As can be seen for low $\frac{E_b}{N_0}$ the spectral efficiency of the ACD is equal to that of the DF-MMSE detector and the SCO detector, since it is preferable to treat adjacent cell interference as noise. However beyond some critical value of $\frac{E_b}{N_0}$, where decoding is preferable, the curves depart and the spectral efficiency of the ACD grows quite rapidly with $\frac{E_b}{N_0}$ (in comparison to the other detectors).

4 Equal Rates

The SCO detector provides equal rates to all users by assigning *equal received powers* to all. With the remaining three detectors, guaranteeing equal rates to all users requires that the SIR function $g(x)$ is a constant. As can be seen from section 3, the above requirement is satisfied for the two linear detectors, i.e, the matched filter detector and the linear MMSE detector, by assigning (again) *equal powers* to all users. This leaves us with the examination of the DF-MMSE detector. We also consider, for the sake of comparison, the DF-MMSE detector which treats adjacent cell interference as additive Gaussian noise (this detector is equivalent to the SCO detector in terms of (overall) spectral efficiency, as mentioned above).

The desired power control function can be numerically obtained[7] by substituting $g(x) = \mathcal{W}$, for some constant $\mathcal{W}$, into either (2-5) or (2-7), and solving these equations so that the power constraint of (2-2) is satisfied.

Results show that in terms of (overall) spectral efficiency, evaluated at the optimum

---

[7] This is done by subdividing the interval $[0, 1]$ into N equal parts, approximating integrals through sums, and using successive approximation methods (our numerical results are based on $N = 25$).

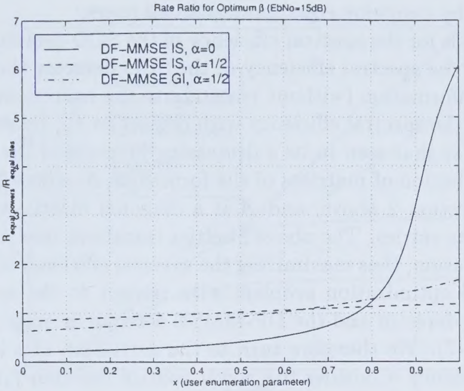


Figure 2: Rate ratio of the equal powers and equal rates strategies for optimum β ($\beta \to \infty$ is represented by $\beta = 10$) and $\frac{E_b}{N_0} = 15dB$. "IS" designates adjacent cell interference suppression, "GI" designates interference treated as noise (both are equivalent for $\alpha = 0$).

choice for β, the equal rates strategy is almost as good as the equal powers strategy, at least for $\frac{E_b}{N_0}$ values of up to $20dB$ (around 98% with interference suppression, closer with the Gaussian interference assumption, worst results for $\alpha = 0$). The spectral efficiency difference is increasing with $\frac{E_b}{N_0}$. However, the equal rates strategy may be preferable in terms of a more equitable rate allocation, as can be seen from the example described by Fig. 2. Fig. 2 shows the ratio between the rate per user while using the DF-MMSE detectors with equal powers, and the rate attained by the equal rates strategy (given by $\log(1 + g(x))/\log(1 + \mathcal{W})$), for $\frac{E_b}{N_0} = 15dB$, $\alpha = 0$, and $\frac{1}{2}$, and optimum choice of β (the latter is very similar for both strategies with interference suppression, less than 1% difference, and identical when interference is treated as noise, thus making the comparison valid). Taking the rate under the equal rates strategy as representing a minimum rate requirement, identical to all users, one can see that with equal *powers* a large proportion of the users fails to meet the requirement. The unbalanced rate distribution is in particular striking for $\alpha = 0$, i.e., in a single cell environment (which should have been expected). For example, with $\frac{E_b}{N_0} = 15dB$ around 76% of the users fall under the rate obtained by the equal rates strategy with $\alpha = 0$. The non-uniformity in rate distribution under the equal powers strategy becomes moderate as we increase the level of adjacent cell interference (results for $\frac{E_b}{N_0} = 20dB$, are of similar in nature). We may therefore conclude that the equal rates strategy is preferable over the equal powers strategy, in terms service quality, when all users are to be equally serviced (as is the case, e.g., with cellular voice communication in current systems). Furthermore, when the $\frac{E_b}{N_0}$ at which the system operates is not too high, choosing the equal rates strategy incures (almost) no loss in overall spectral efficiency.

5 Maximum Spectral Efficiency

Observing (2-4), it is quite straightforward to show, using Jensen's inequality and the convexity of the $\log(1 + x)$ function, that the spectral efficiency of the matched filter detector is maximized by assigning equal powers to all users.

The same result holds for the spectral efficiency of the SCO detector. This is observed by direct evaluation of the spectral efficiency of the SCO detector through the channel's input-output mutual information (without resorting to the equivalent DF-MMSE detector). The derivative of the spectral efficiency with respect to $\bar{P}_{\mathrm{eq}}$ (taken as an argument), as defined in section 3, is thus seen to be a decreasing function of the Stieltjes transform of the eigenvalue distribution of matrices of the form $\boldsymbol{S}_i^{\dagger}\boldsymbol{S}_i\boldsymbol{A}$, where $\boldsymbol{S}_i$ is the signatures matrix as defined in section 2 above, and $\boldsymbol{A}$ is a diagonal matrix with the normalized powers of the users as its entries. The above Stieltjes transform (see [10]) is minimized by the equal power distribution, thus maximizing the spectral efficiency of the SCO detector.

Solving the general optimization problem with respect to the spectral efficiency of both the linear MMSE detector and the DF-MMSE detector is quite complex, as can be seen from (2-6) and (2-7). We therefore turn to the derivation of a lower bound on the maximum spectral efficiency. Consider the power control function $f_\rho(x)$ defined as

$$f_\rho(x) = \begin{cases} \frac{\bar{P}}{\rho} & 0 < x \leq \rho \\ 0 & \text{otherwise} \end{cases}, \quad 0 < \rho \leq 1\,. \tag{5-1}$$

Clearly, $f_\rho(x)$ satisfies the average power constraint (see (2-2)). The rationale behind $f_\rho(x)$ is that given a system, with some fixed $\frac{E_b}{N_0}$, interference factor α, and system load β, one might gain in spectral efficiency by effectively "shutting down" part of the users' transmissions by means of the power control function, and assigning more power to the remaining users while satisfying the average power constraint[8]. Obviously, assigning all power to a single user ($\rho \to 0$) is useless since this drives the system's spectral efficiency to zero. The power control function $f_\rho(x)$ comes into effect if there exists some value of $\rho = \rho^* < 1$ for which the spectral efficiency is maximized. In the case in which $\rho^* = 1$, $f_\rho(x)$ simply reduces to the equal powers strategy discussed in section 3. The above optimization is in fact equivalent to the optimization of the spectral efficiency with respect to β under the equal powers strategy, for fixed $\frac{E_b}{N_0}$, if the initially set value of β is higher than the optimum value (note that $f_\rho(x)$ cannot *increase* the effective load). Hence, for a given system, at a fixed working point specified by $\frac{E_b}{N_0}$, α, and β, the maximum spectral efficiency, where maximization is taken over the set of power control functions, is lower bounded by the spectral efficiency obtained by substituting $f_\rho(x)$ into (2-6) and (2-7), and evaluating the spectral efficiency at $\rho = \rho^*$.

Some comparative results, demonstrating the effect of maximizing the spectral efficiency via the power control function are demonstrated in Fig. 3. Here the spectral efficiency of all four detectors is plotted for $\alpha = \frac{1}{2}$, and $\beta = 1$. The spectral efficiency of the linear MMSE and DF-MMSE detector is evaluated assigning the power control function $f_\rho(x)$, with $\rho = \rho^*$. For the matched filter detector and the SCO detector we evaluated the spectral efficiency with the equal power assignment (which attains the maximum). Also included in Fig. 3 are the spectral efficiencies without random spreading of the three detectors mentioned in section 3 (denoted by "N.S."). As can be seen, since $\beta = 1$, for low $\frac{E_b}{N_0}$ the spectral efficiencies of the linear MMSE detector and the DF-MMSE detector, are lower than the spectral efficiencies without random spreading of the matched filter and SCO detector, respectively (equality is attained for $\beta \to \infty$). This is also the case with the spectral efficiency of the matched filter, and the SCO detector, for

---

[8] Notice, however, that using the function $f_\rho(x)$ still confines us to equal power assignment to all effectively active users.

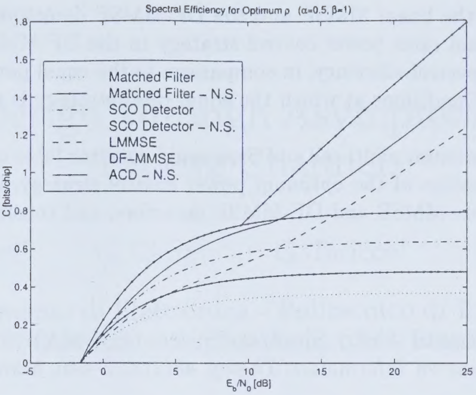


Figure 3: Spectral efficiency comparison using $f_\rho(x)\big|_{\rho=\rho^*}$, for $\alpha=\frac{1}{2}$, and $\beta=1$.

all values of $\frac{E_b}{N_0}$. However, beyond some critical values of $\frac{E_b}{N_0}$ the optimum β for the linear MMSE and the DF-MMSE detectors decreases below unity, and thus using the power control function $f_\rho(x)$ (with $\rho=\rho^*$) brings the spectral efficiency of both detectors to its maximum value with the equal power assignment (c.f. Fig. 1).

6 Summary and Conclusions

In this paper we analyzed the spectral efficiency of four multiuser detectors, which differ by the amount and type of information made available to the detector, with regard to interference (both intra-cell and inter-cell). Our results demonstrate the dramatic effect of information about interfering signals, which is most clearly seen by comparing the linear MMSE detector and the SCO detector (the DF-MMSE detector is obviously superior to all, being the most informed, and the matched filter detector is inferior to all, being the least informed). We showed that one can gain, even without trying to decode the transmissions of the interfering users in adjacent cells[9], or treating them optimally in the setting of an interference channel (see [5]), by the very fact that the linear MMSE filter will account for reducing the interference they cause, provided that their signatures are known. It was shown that when $\frac{E_b}{N_0}$ and the interference factor are not too low, the relatively simple *linear* MMSE detector, which is more informed regarding adjacent cell interference, is preferable over the SCO detector which employs *"optimum"* detection of intra-cell users, while treating inter-cell interference as noise. We may therefore conclude that for high data rates, inherently demanding high $\frac{E_b}{N_0}$, the MMSE based detection strategies yield a pronounced gain over the other two strategies considered. The penalty due to random spreading was also demonstrated.

Comparing power control strategies we showed that the equal power assignment obtains equal rates with the matched filter detector, the SCO detector, and the linear MMSE detector, and it attains the maximum spectral efficiency with the matched filter and the SCO detectors. This strategy may also provide a lower bound to the maximum spectral efficiency of the linear MMSE and the DF-MMSE detectors. Our results show that applying the equal rates power control strategy to the DF-MMSE detector results in almost no loss in spectral efficiency, in comparison to the equal powers strategy (when $\frac{E_b}{N_0}$ is not too high). Conditions at which the equal rates strategy is preferable were also discussed.

---

[9] Which enables interference *cancellation*.

Analysis of the optimum multi cell site processing detector [4] is under current study, as well as characterization of the *optimum* power control strategy, in terms of spectral efficiency, for the linear MMSE and DF-MMSE detectors, and consideration of fading.

References

[1] S. Verdú and S. Shamai (Shitz). Spectral efficiency of CDMA with random spreading. *IEEE Transactions on Information Theory*, 45(2):622–640, March 1999.

[2] D. Tse and S. Hanly. Linear multiuser receivers: Effective interference, effective bandwidth and user capacity. *IEEE Transactions on Information Theory*, 45(2):641–657, March 1999.

[3] S. V. Hanly and D. N. C. Tse. Resource pooling and effective bandwidths in CDMA networks with multiuser receivers and spatial diversity. *A Report*, May 16 1999.

[4] A. D. Wyner. Shannon-theoretic approach to a Gaussian cellular multiple-access channel. *IEEE Transactions on Information Theory*, 40(6):1713–1727, November 1994.

[5] S. Shamai (Shitz) and A. D. Wyner. Information-theoretic considerations for symmetric, cellular, multiple-access fading channels - Parts I & II. *IEEE Transactions on Information Theory*, 43(6):1877–1911, November 1997.

[6] H. V. Poor and S. Verdú. Probability of error in MMSE multiuser detection. *IEEE Transactions on Information Theory*, 43(3):858–871, May 1997.

[7] J. Zhang, E. K. P. Chong, and D. N. C. Tse. Output MAI distributions of linear MMSE multiuser receivres in CDMA systems. Technical report, Purdue University - UC Berkeley, 1999.

[8] M. K. Varanasi and T. Guess. Achieving vertices of the capacity region of the synchronous Gaussian correlated-waveform multiple-access channel with decision feedback receivers. In *Proceedings of ISIT'97*, page 270, Ulm, Germany, June 29 – July 4, 1997.

[9] B. M. Zaidel, S. Shamai (Shitz), and S. Verdú. Spectral efficiency of randomly spread DS-CDMA in a multi-cell model. Report, Technion - Israel Institute of Technology, 1999.

[10] J. W. Silverstein. Strong convergence of the empirical distribution of eigenvalues of large dimensional random matrices. *Journal of Multivariate Analysis*, 55:331–339, 1995.

[11] S. Shamai (Shitz) and S. Verdú. Capacity of CDMA fading channels. In *Proceedings of ITW'99*, page 24, Metsovo, Greece, June 27 – July 1, 1999.

CDMA Design Through Asymptotic Analysis: Fading Channels

E. Biglieri[†] G. Caire[‡] G. Taricco[†] E. Viterbo[†]

† Dipartimento di Elettronica – Politecnico di Torino – Italy
‡ Eurecom – Sophie Antipolis – France

Abstract

Using asymptotic analysis, we show how flat fading affects CDMA with linear receivers. Specifically, we let the number of users grow without bound while the ratio of number of users to spreading sequence length is kept fixed to a value α. *We treat separately the case of flat, slow fading with lognormal shadowing (nonergodic channel) and of flat, fast Rayleigh fading (ergodic channel). For the former channel we derive the outage probability, while for the latter we compute the channel capacity. We show in particular that, with fading, as* $\alpha \to \infty$ *the system capacity tends to the same limit of 1.44 bit/symbol as for the non-fading channel. On the contrary, the outage probability exibits a floor for all* α *values when a single-user matched filter is employed, while with a linear minimum-mean-square-error receiver the floor is present only for* $\alpha > 1$.

1 Introduction

We examine a CDMA system with error-control coding operating on a channel affected by flat fading. We treat separately the case of flat, slow fading with lognormal shadowing (nonergodic channel) and of flat, fast Rayleigh fading (ergodic channel). For the former channel we derive the outage probability, while for the latter we compute the channel capacity. The receiver consists of a linear front-end, viz., either a single-user matched filter (SUMF) detector or a linear minimum-mean-square error receiver (MMSE) detector [6], followed by a single-user decoder. The key performance measure here is the Signal-to-Interference plus Noise Ratio (SINR) at the output of the linear filter: users' quality of service can be expressed in terms of a target SINR. The SINR cumulative distribution function (cdf) yields immediately the *outage probability*, i.e., the probability that the actual SINR, say β, is below the required SINR target, denoted β_0. In general, β_0 depends on the coding scheme. Our study is asymptotic, in the sense that the number of users grows without bound, while the ratio of number of users to spreading-sequence length is kept fixed. In this work, after a description of the system model, we shall examine the outage probability of slow-fading channels in Section 3, and the capacity of fast-fading Rayleigh channel in Section 4.

2 System model

We consider a single-cell, chip-synchronous DS-CDMA system. Our model involves K users, random spreading sequences, and a spreading-sequence length $L = T/T_c$, where $1/T$ is the symbol rate and $1/T_c$ the chip rate. We assume (as in [3, 5]) a large number of users ($K \to \infty$) and $K/L \to \alpha$ (a constant "channel load" as the length of the spreading sequences increases to accommodate the K users). Let the empirical distribution function of the received interfering power from all users converge, as $K \to \infty$, to $F_{\mathcal{P}}(x)$. Specifically, denote by $\mathcal{P}_k$ the power received from user k, and by $F_K(x)$ the empirical cumulative distribution function of the received powers:

$$F_K(x) = \frac{1}{K} \sum_{k=1}^{K} \Im\{\mathcal{P}_k \leq x\}$$

where $\Im\{\mathcal{A}\}$ is the indicator function of the event $\mathcal{A}$. We require that $F_K(x) \to F_{\mathcal{P}}(x)$ almost everywhere as $K \to \infty$. Thus, $F_{\mathcal{P}}(x)$ is the average cumulative distribution function of the power received from a user picked at random (with uniform probability $1/K$) from the population of $K \to \infty$ users. In our previous paper [2] we chose for $F_{\mathcal{P}}(x)$ the step function

$$F_{\mathcal{P}}(x) = u(x - \mathcal{P}_0)$$

corresponding to a situation in which the power received from all users is $\mathcal{P}_0$. From [3, 4], with single-user matched filter (SUMF) reception and under mild convergence conditions we have the asymptotic SINR:

$$\beta = \frac{\mathcal{P}}{N_0 + \alpha \int_0^\infty x \, dF_{\mathcal{P}}(x)}, \tag{1}$$

where $\mathcal{P}$ is the useful power received from reference user, and N_0 is the variance of the additive white Gaussian noise samples. With linear minimum mean-square error (MMSE) reception the asymptotic SINR β is the unique real nonnegative solution of

$$\beta = \frac{\mathcal{P}}{N_0 + \alpha \displaystyle\int_0^\infty \frac{x\mathcal{P}}{\mathcal{P} + x\beta} \, dF_{\mathcal{P}}(x)}. \tag{2}$$

2.1 Methodological premises

The intricacies of dealing with fading channels are thoroughly described in [1], so we shall not delve into them here. Rather, we list a number of points that describe the rationale behind the calculations that follow. In particular, we motivate the different methods that are used to analyze the two situations of flat, slow fading and of flat, fast fading.

2.1.1 Flat, slow fading

Consider first flat, slow fading—not slow enough to be compensated by the power-control system, yet slow enough to make the channel nonergodic. The fading is due to lognormal shadowing. This situation occurs for example when the power control is imperfect, thus leaving a residual shadowing of 2 dB (say), rather than the 8 dB of the uncompensated shadowing. In this situation, (1) and (2) are still valid, provided that in lieu of $\mathcal{P}$ we write $z_1\mathcal{P}$, where z_1 is the lognormal RV modeling the shadowing in user-1 channel, and in lieu of $F_{\mathcal{P}}$ we write the limiting empirical distribution of the product $z_k\mathcal{P}_k$, where z_k is the lognormal RV modeling user-k channel and $\mathcal{P}_k$ is the power received from user k when there is no shadowing. Even if we choose all the $\mathcal{P}_k$ equal (to $\mathcal{Q}$, say), the distribution $F_{\mathcal{P}}$ is not a step function, but rather equals the distribution of the lognormal RV $z\mathcal{Q}$, where z is a generic lognormal RV. Moreover, since we have $z_1\mathcal{P}$ instead of $\mathcal{P}$, the value of β becomes a function of z_1, and hence a random variable. We will see in the following that β is proportional to z for both SUMF and MMSE. We may define an outage probability

$$P(\beta < \beta_0) \tag{3}$$

which will not be generally a step function as in the nonfaded channel [2], even in our limiting case of K and L both growing to infinity. The value of β_0 is chosen as in [2]. For example, if we use a nonideal code with rate R bit/symbol achieving the target performance at a certain E_b/N_0, we set $\beta_0 = RE_b/N_0$. If we consider instead an optimum code which operates at the Shannon limit for a Gaussian channel and we want a rate R, then we set

$$\beta_0 = 2^R - 1 \tag{4}$$

as we did in [2].

2.1.2 Flat, fast fading

This is the case of a fading so fast that it cannot be compensated by power control but constant over one symbol, and such as to make the channel ergodic. In this situation, (1) and (2) are still valid under the same modification as in the previous subsection, but now z_1 and z are distributed as the instantaneous power gain of the channel. Here we choose $\sqrt{z}$ and $\sqrt{z_1}$ to be Rayleigh-distributed, so that z_1 and z are exponentially distributed (with mean 1). Here the value of β (which, being a function of z_1, is a random variable) can be interpreted as the instantaneous SINR at the output of the linear receiver. With optimum (Gaussian) codes, the capacity is given by [1]

$$C = \mathbb{E}[\log_2(1+\beta)] \tag{5}$$

where the expectation $\mathbb{E}$ is taken with respect to the distribution of β. Eq. (5) can be evaluated by using numerical techniques. With SUMF, β is still exponentially distributed. With MMSE, we can expect the distribution of β to be highly complex, so that the average capacity will be even worse (there is no conceptual complication, though). A major departure from the situation of previous subsection occurs with the definition of outage probability: in fact the probability $P(\beta < \beta_0)$ has no practical significance anymore (although a widespread misconception leads it sometimes to be interpreted as an outage probability). This is due to the fact that β is the instantaneous SINR, and with coding and fast fading the probability of error does not depend on the instantaneous SINR (which is a RV), but rather on the statistics of SINR and on the code features, and is difficult to compute. We argue that in our case the sensible quantity is the *system capacity*, defined as

$$C_{\text{sys}} \triangleq \max \alpha C(\alpha) \quad \text{bit/s/Hz} \tag{6}$$

and we write $C(\alpha)$ rather than C to stress the fact that the capacity depends on α. If we want all users to transmit reliably at rate R, the maximum admissible α is

$$\alpha_{\max} \triangleq \max\{\alpha : R \leq C(\alpha)\} \tag{7}$$

In this way, we can still define a *system outage probability*: this is now a step function, taking on value 0 if $\alpha < \alpha_{\max}$, and 1 if $\alpha > \alpha_{\max}$.

3 Slow-fading channel

Let $\eta \triangleq \mathcal{P}/N_0 = RE_b/N_0$ be the signal-to-noise ratio for the single user.

3.1 SUMF receiver

With the SUMF receiver,

$$\beta = \frac{z_1\mathcal{P}}{N_0 + \alpha\mathcal{P}\mathbb{E}[z]} = \frac{z_1\mathcal{P}}{N_0 + \alpha\mathcal{P}}$$

where z_1 and z denote the power gains of the useful and interfering users, respectively, and follow a log-normal distribution with log-standard deviation σ dB and mean value $\mathbb{E}[z] = 1$. In other words,

$$10 \log_{10} z \sim \mathcal{N}(-\mu_z, \sigma)$$

where $\mu_z \triangleq \sigma^2 (\ln 10)/20$. Letting $\kappa_s(\alpha, \eta) \triangleq 1/(\eta^{-1} + \alpha)$ the outage probability is given by

$$\begin{aligned} P_{\text{out,s}} &= P(\beta \le \beta_0 = 2^R - 1) = P\left(z_1 \le (2^R - 1)/\kappa_s(\alpha, \eta)\right) \\ &= 1 - Q\left[\sigma^{-1}\left(\mu_z + 10 \log_{10}[(2^R - 1)/\kappa_s(\alpha, \eta)]\right)\right] \end{aligned} \tag{8}$$

where $Q(x) \triangleq P(\mathcal{N}(0,1) > x)$.

3.2 MMSE receiver

With the MMSE receiver, the asymptotic SINR is the solution of the following equation, obtained from the substitutions $\mathcal{P} \to z_1\mathcal{P}$ and $x \to z\mathcal{P}$ in (2):

$$\beta = \frac{z_1\mathcal{P}}{N_0 + \alpha\mathcal{P}\mathbb{E}\left[\dfrac{z_1 z}{z_1 + \beta z}\right]} \tag{9}$$

where $\mathbb{E}[\cdot]$ denotes expectation with respect to z. This equation can also be expressed as follows:

$$z_1/\beta = \eta^{-1} + \alpha\mathbb{E}_z\left[\frac{z}{z_1/\beta + z}\right](z_1/\beta) \tag{10}$$

and can be solved iteratively by the recursion

$$\begin{aligned} \xi_n &= \eta^{-1} + \alpha\mathbb{E}\left[\frac{z}{\xi_{n-1} + z}\right]\xi_{n-1} \\ &= \eta^{-1} + \frac{\alpha}{\sqrt{2\pi}}\xi_{n-1}\int_{-\infty}^{\infty}\frac{e^{-x^2/2}}{\xi_{n-1}10^{(\mu_z - \sigma x)/10} + 1}\,dx \end{aligned}$$

initialized by $\xi_0 = 1$. This iteration converges to a constant $\lim_{n\to\infty}\xi_n = \kappa_m(\alpha, \eta)$ as shown in the following subsection which is the solution ξ of eq. (10). Its solution can be written in the form

$$\beta = \kappa_m(\alpha, \eta)z_1 \tag{11}$$

Note that β is proportional to z_1 by a deterministic factor depending only on the parameters α and η. The outage probability is

$$P_{\text{out,m}} = P\left(z_1 \leq (2^R - 1)/\kappa_{\text{m}}(\alpha, \eta)\right) = 1 - Q\left[\sigma^{-1}\left(\mu_z + 10\,\log_{10}[(2^R - 1)/\kappa_{\text{m}}(\alpha, \eta)]\right)\right]$$

3.3 Outage probability floor

The above results show that with both receivers the outage probability may approach a nonzero limit as $\eta \to \infty$ (outage probability floor).

With the SUMF receiver, since $\lim_{\eta\to\infty} \kappa_{\text{s}}(\alpha, \eta) = \alpha^{-1}$, from (8) the outage probability floor is

$$P_{\text{out,s}} = 1 - Q\left[\sigma^{-1}\left(\mu_z + 10\,\log_{10}[(2^R - 1)\alpha]\right)\right] \tag{12}$$

for any value of α. This result derives from the fact that the SUMF receiver is not near-far resistant [6]: because of the fading, with nonzero probability some interferer is strong enoughto make the SINR fall below the outage threshold.

As for the MMSE receiver, we have an outage probability floor if and only if $\alpha > 1$, i.e., the number of users exceeds the spreading factor. In fact, letting $\eta \to \infty$ in (10) we obtain the following equation:

$$\alpha \mathbb{E}_z\left[\frac{\beta z}{z_1 + \beta z}\right] = 1$$

This equation has a solution if and only if $\alpha \geq 1$: in fact, the second factor in its LHS is always less than or equal to 1. This result is due to the lack of near-far resistance of the MMSE receiver when $\alpha > 1$ [6].

The outage probability floor is illustrated in Fig. 1, obtained by plotting P_{out} versus E_b/N_0 with the MMSE and SUMF receivers, $\alpha = 0.2, 0.5, 0.8, 1, 1.2, 1.5$, rates $R = 1$ and 2 bit/symbol, and $\sigma = 2$ and 8 dB. The outage probability degrades, as we see, by increasing either R, α, or σ, which represent the user rate, system load, and shadowing level, respectively.

4 Flat, fast fading

The channel capacity is given by

$$C(\alpha) = \mathbb{E}[\log_2(1 + \beta)]$$

Here, z and z_1 are exponentially-distributed RV's with unit mean. For the SUMF receiver, we get

$$C_{\text{s}}(\alpha) = \int_0^{\infty} \log_2\left(1 + \frac{z_1 \mathcal{P}}{N_0 + \alpha\mathcal{P}}\right) e^{-z_1} dz_1 = e^{1/\kappa_{\text{s}}(\alpha,\eta)}\,\text{Ei}(1, 1/\kappa_{\text{s}}(\alpha, \eta)) \log_2 e$$

For the MMSE receiver, equation (10) becomes

$$\begin{aligned} z_1/\beta &= \eta^{-1} + \alpha\,(z_1/\beta) \int_0^{\infty} \frac{z}{z_1/\beta + z} e^{-z}\, dz \\ &= \eta^{-1} + \alpha\,(z_1/\beta)[1 - (z_1/\beta)e^{z_1/\beta}\text{Ei}(1, z_1/\beta)] \end{aligned} \tag{13}$$

which yields $\beta = \kappa'_{\mathrm{m}}(\alpha, \eta) z_1$. Then, the capacity is given by

$$C_{\mathrm{m}}(\alpha) = \int_0^\infty \log_2[1 + \kappa'_{\mathrm{s}}(\alpha, \eta) z_1] e^{-z_1} dz_1 = e^{1/\kappa'_{\mathrm{m}}(\alpha,\eta)} \operatorname{Ei}(1, 1/\kappa'_{\mathrm{m}}(\alpha, \eta)) \log_2 e \qquad (14)$$

Figure 5 shows the capacity curves for SUMF and MMSE as a function of $\mathcal{P}/N_0 = \eta$ for different values of α. Note that the capacity is always bounded for the SUMF receiver and for the MMSE receiver when $\alpha > 1$.

4.1 System capacity

We define the system capacity as

$$C_{\mathrm{sys}} = \max_{\alpha : R \leq C(\alpha)} \alpha\, C(\alpha)$$

Figure 3 shows both $C(\alpha)$ and $\alpha\, C(\alpha)$ versus α at $\eta = 6$, 10, 20 and 30 dB, respectively. They illustrate the following results.

- With the SUMF receiver, the product $\alpha\, C(\alpha)$ increases with α, while for the MMSE receiver it exhibits a maximum.

- As $\alpha \to \infty$, with both receivers $\alpha\, C(\alpha)$ approaches the same limit, 1.44 bit/symbol, regardless of η. In fact, with the SUMF receiver we have

  $$\lim_{\alpha \to \infty} \alpha\, C_{\mathrm{s}}(\alpha) = \lim_{\alpha \to \infty} \alpha\, e^{\eta^{-1} + \alpha} \operatorname{Ei}(1, \eta^{-1} + \alpha) \log_2 e = \log_2 e = 1.44 \text{ bit/symbol}$$

  With the MMSE receiver, when $\alpha \to \infty$, we have from (13):

  $$\xi = \eta^{-1} + \alpha\xi[1 - \xi\, e^{\xi} \operatorname{Ei}(1, \xi)] = \alpha\xi[1/\xi + O(1/\xi^2)] = \alpha[1 + O(1/\xi)]$$

  where we set $\xi = z_1/\beta$. Now, from $\xi = \alpha[1 + O(1/\xi)]$ we have $\xi = \alpha[1 + q/\xi]$ for some finite q when both ξ and α are sufficiently large. Thus, $\xi^2 - \alpha\xi - q = 0$, which can be solved as follows:

  $$\xi = \frac{1}{2}[\alpha + \sqrt{\alpha^2 + 4q}] = \frac{1}{2}[\alpha + \alpha\sqrt{1 + 4q/\alpha^2}] = \frac{1}{2}[\alpha + \alpha + 2q/\alpha + O(\alpha^{-3})]$$

  It follows that $\xi = \alpha + O(1/\alpha)$. Hence

  $$\lim_{\alpha \to \infty} \alpha\, C_{\mathrm{m}}(\alpha) = \lim_{\alpha \to \infty} \alpha\, e^{\xi} \operatorname{Ei}(1, \xi) \log_2 e = \lim_{\alpha \to \infty} \alpha[1/\xi + O(1/\xi^2)] \log_2 e = 1.44$$

  Since in a non-faded channel both SUMF and MMSE achieve the same limiting spectral efficiency of 1.44 bit/s/Hz [2], we have shown that fast Rayleigh fading does not reduce system capacity.

- The maximum system capacity with SUMF is achieved for $\alpha \to \infty$. On the contrary, with MMSE receivers an optimal $\alpha = \alpha_{\mathrm{opt}}$ exists which maximizes $\alpha\, C_{\mathrm{m}}(\alpha)$; thus, the system capacity is $\alpha_{\mathrm{opt}}\, C_{\mathrm{m}}(\alpha_{\mathrm{opt}})$. This optimum can be found by solving (13) for α:

  $$\alpha = \frac{\xi - \eta^{-1}}{\xi[1 - \xi\, e^{\xi} \operatorname{Ei}(1, \xi)]}$$

which yields

$$\alpha\, C_{\mathrm{m}}(\alpha) = \frac{\xi - \eta^{-1}}{\xi[1 - \xi\, e^{\xi}\, \mathrm{Ei}(1,\xi)]} e^{\xi}\, \mathrm{Ei}(1,\xi) \log_2 e$$

The optimum ξ can be obtained and Fig. 4 shows the system capacity and the optimal load versus the SNR η. The system capacity is increasing asymptotically linearly with the SNR. Note that α_{opt} has a minimum at SNR=30.9 dB (0.8635). This fact can be explained by noting that

1. For low SNR, the optimum linear MMSE receiver approaches the SUMF receiver since noise dominates MAI. In this case, system capacity is maximum when $\alpha \to \infty$, as with the SUMF receiver (additionally, the SNR$\to 0$).
2. For high SNR, the optimum linear MMSE receiver approaches the decorrelating receiver which is the optimal linear receiver in the absence of noise. In this case, system capacity is maximum for $\alpha \to 1$. In fact, if SNR$= \eta \to \infty$,

   $$\alpha C_{\mathrm{m}}(\alpha) = e^{\xi} \frac{\mathrm{Ei}(1,\xi)}{1 - \xi e^{\xi} \mathrm{Ei}(1,\xi)} \log_2 e$$

   The rhs is a monotonically decreasing function of ξ so that the maximum is attained when $\xi = 0$ and

   $$\lim_{\xi \to 0} \alpha = \lim_{\xi \to 0} \frac{1}{1 - \xi e^{\xi} \mathrm{Ei}(1,\xi)} = 1$$

Then, the continuity of $\alpha_{\mathrm{opt}}(\eta)$ implies that a maximum (≤ 1) exists.

4.1.1 Gaussian channel

For comparison, let us consider the channel capacity in the case of a purely Gaussian channel without fading where all users transmit with the same power $\mathcal{P}$ [3, 5]

$$C(\alpha) = \log_2(1 + \beta)$$

For the SUMF receiver the SINR is simply

$$\beta = \frac{1}{\eta^{-1} + \alpha}$$

For the MMSE receiver the SINR can be evaluated from

$$\beta = \frac{1}{\eta^{-1} + \frac{\alpha}{1+\beta}}$$

which yields

$$\beta = \frac{1}{2}\left[\eta(1-\alpha) - 1 + \sqrt{1 + 2\eta(1+\alpha) + \eta^2(1-\alpha)^2}\right]$$

Figure 5 show $C(\alpha)$ and $\alpha C(\alpha)$ for the two receivers for different $\mathcal{P}/N_0$. We observe that the Gaussian channel capacity is slightly higher than that of the fading channel, but for $\alpha \to \infty$ both converge to the same limit of 1.44 bit/symbol.

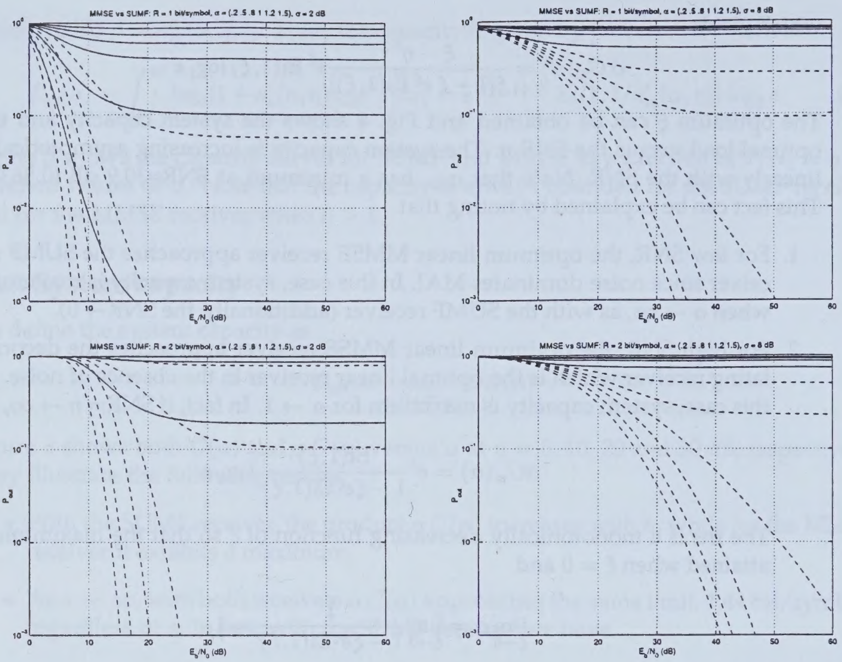


Figure 1: Outage probability with MMSE (dashed) and SUMF (solid) for rates $R = 1$ and 2 bit/symbol, $\sigma = 2$ and 8 dB, and $\alpha = 0.2, 0.5, 0.8, 1, 1.2, 1.5$.

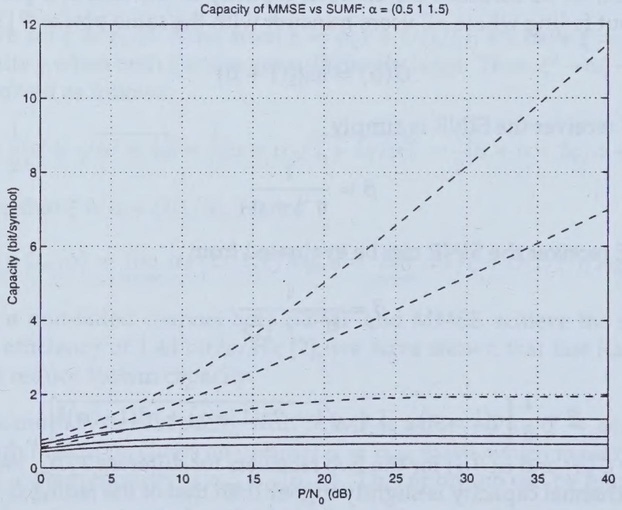


Figure 2: Capacity of MMSE (dashed) and SUMF (solid) (bit/symbol), $\alpha = 0.5, 1, 1.5$.

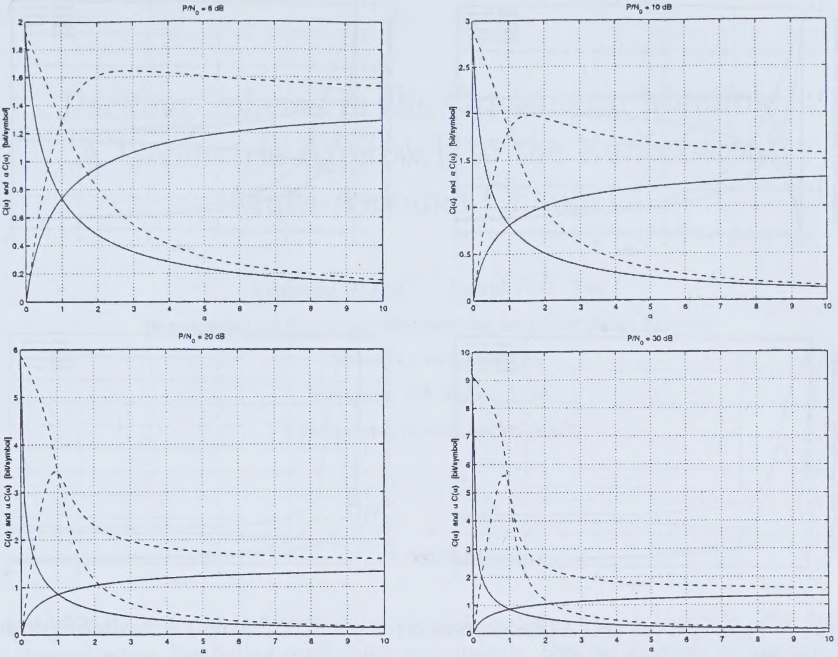


Figure 3: Plot of $C(\alpha)$ and $\alpha\, C(\alpha)$ versus α for the MMSE (dashed) and SUMF (solid) receiver for $\eta = 6, 10, 20, 30$ dB.

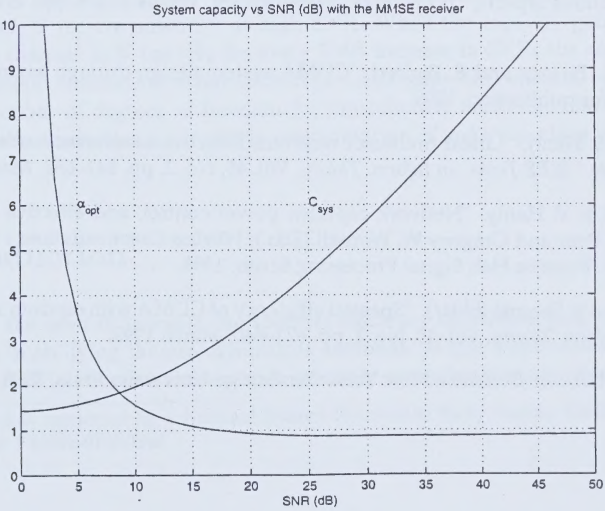


Figure 4: System capacity C_{sys} and optimal α versus η for the MMSE receiver.

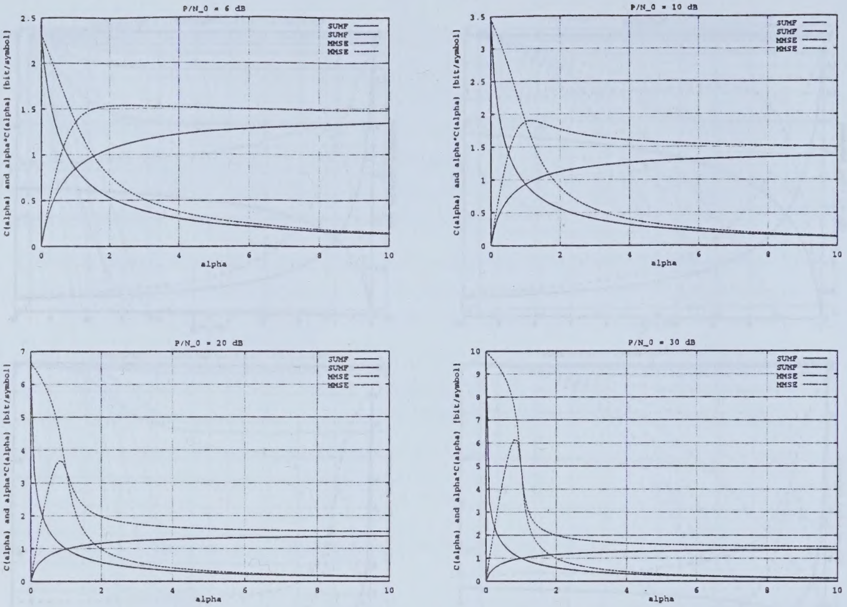


Figure 5: Plots of $C(\alpha)$ and $\alpha\, C(\alpha)$ versus α for the SUMF (solid) and MMSE (dashed) receiver for $\eta = 6, 10, 20, 30$ dB.

References

[1] E. Biglieri, J. Proakis, and S. Shamai (Shitz), "Fading channels: Information-theoretic and communications aspects," *IEEE Trans. Inform. Theory*, Vol. 44, No. 6, pp. 2619–2692, October 1998.

[2] G. Caire, G. Taricco, and E. Biglieri, "CDMA system design through asymptotic analysis," submitted for publication, 1999.

[3] D. Tse and S. Hanly, "Linear multiuser receivers: Effective interference, effective bandwidth and capacity," *IEEE Trans. on Inform. Theory*, Vol. 45, No. 2, pp. 641–657, March 1999.

[4] D. Tse and S. V. Hanly, "Network capacity, power control, and effective bandwidth," in: H. Vincent Poor and Gregory W. Wornell (Eds.), *Wireless Communications : Signal Processing Perspectives*. Prentice Hall Signal Processing Series, 1998.

[5] S. Verdú and S. Shamai (Shitz), "Spectral efficiency of CDMA with random spreading," *IEEE Trans. on Inform. Theory*, Vol. 45, No. 2, pp. 622–640, March 1999.

[6] S. Verdú, *Multiuser Detection*. New York: Cambridge University Press, 1998.

Packing Spheres in the Grassmann Manifold: A Geometric Approach to the Noncoherent Multi-Antenna Channel *

Lizhong Zheng David N.C. Tse
Department of Electrical Engineering and Computer Sciences
University of California
Berkeley, CA 94720, USA
{lzheng,dtse}@eecs.berkeley.edu

Abstract

In this paper, we study the capacity of multiple antenna fading channels. We focus on the channel where the fading coefficients vary quickly, thus an accurate estimation of the coefficients are generally not available to either the transmitter or the receiver. We use a noncoherent model proposed by Marzetta and Hochwald. The model does not assume any prior knowledge at the receiver nor the transmitter of the instantaneous fading coefficients, but only assumes that they remain constant for a coherent interval T as an approximation of the continuously varying channel. We compute explicitly the capacity of this channel with N transmit and N receive antennas at high SNR. While the capacity gain of the coherent multi-antenna channel is N bps/Hz for every 3 dB increase in SNR, the corresponding gain for the noncoherent channel turns out to be $K(1-K/T)$ bps/Hz, where $K = \min\{N, \lfloor T/2 \rfloor\}$. This is the number of degrees of freedom for noncoherent multi-antenna communications. The capacity expression has a geometric interpretation of *sphere packing in the Grassmann manifold.*

1 Introduction

Motivated by the need to increase the spectral efficiency of wireless systems, a major effort is being made in studying the use of multiple antennas. While much work has been done on systems with multiple *receive* antennas, it was only recently shown by Foschini and Telatar

---

*This research is supported by a National Science Foundation Early Faculty CAREER Award and by DARPA grant # F30602-97-2-0346

[1, 2, 3] that much larger spectral efficiency can be achieved by utilizing spatial diversity at *both* the transmitter and the receiver.

In a single-antenna additive white Gaussian noise (AWGN) channel, it is well known that at high signal-to-noise ratio (SNR), 1 bit/s/Hz (bps/Hz) capacity gain can be achieved with every 3dB increase in SNR. In contrast, for a multiple antenna system with N transmit and N receive antennas and i.i.d. Rayleigh fading between all antenna pairs, the capacity gain is N bps/Hz for every 3dB SNR increase [2]. This suggests the potential for very sizable improvement in spectral efficiency.

The result above is derived under the key assumption that the instantaneous fading coefficients are known to the receiver. Thus, this result can be viewed as a fundamental limit for *coherent* multi-antenna communications. In fixed wireless environment, the fading coefficients vary slowly, so the transmitter can periodically send pilot signals to allow the receiver to estimate the coefficients accurately. However, in mobile environments, the fading coefficients can change quite rapidly and the estimation of channel parameters becomes difficult, particularly in a system with large number of antenna elements. In this case, there may not be enough time to estimate the parameters precisely enough. Also, the time one spends in sending pilot signals is not negligible, and the tradeoff between sending more pilot signals to estimate the channel more accurately and using more time to communicate to get more data through becomes an important factor affecting the performance. In such situations, one may also be interested in exploring schemes that do not need explicit estimates of the fading coefficients. It is therefore of interest to understand the fundamental limits of *non-coherent* multi-antenna communications.

A line of work was initiated by Marzetta and Hochwald [4, 5] to study the capacity of multi-antenna channels when neither the receiver nor the transmitter knows the fading coefficients of the channel. They used a channel model where the fading gains are i.i.d. Rayleigh and remain constant for T symbol periods before changing to new independent realizations. Under this assumption, they reached the conclusion that further increasing the number of transmit antennas N beyond T cannot increase the capacity. They also characterized certain structure of the optimal input distribution, and computed explicitly the capacity of one transmit and one receive antenna case at high SNR.

In this paper, we will use the same model to study the channel capacity for general $N > 1$ transmit and receive antennas.We will focus on the high SNR regime, of interest particularly for narrow band applications but also seems more tractable than the general problem. The high SNR capacity for the single-antenna case is obtained in [5] from first principles, by direct analysis of the integral involved in the relevant mutual information functional. It seems difficult to generalize their technique to the multi-antenna case. Using a geometric approach, we transformed the problem into a new coordinate system which describes the underlying geometry more naturally and in which the input optimization problem can be easily solved. We obtain an explicit formula for the high SNR channel capacity with $T \geq 2N$, as well as characterize the rate that capacity increases with SNR for $T < 2N$. We define $K = \min(N, \lfloor T/2 \rfloor)$, and conclude that the capacity gains $K(1-K/T)$ bps/Hz for every $3dB$ SNR increase. A geometric interpretation of the result as "Packing Spheres in Grassmann manifold" is also given at the end of the paper.

We will outline the main ideas in the proofs of the results, while relegating the complete proofs to a full version of this paper.

2 System Model and Basic Results

2.1 System Model

We follow the model in [5]. Assume the system has N transmit and N receive antennas, with i.i.d. Gaussian noise at each of the receive antennas. The propagation coefficients form a $N \times N$ random matrix which neither the transmitter nor the receiver knows. We adopt a Rayleigh fading model. We also assume that the coefficients remain constant for a time period T, and change to new independent realizations in the next time period. This model can be interpreted in several ways:

- as an approximate model for a continuous fading channel with coherence time T;
- as a model for frequency hopping, ideally interleaved TDMA or packet-based system where each frame of data sees an independent realization of the channel but the channel is constant within each frame;
- as a model for non-coherent communications where the receiver assumes that the channel remains constant over a certain number of symbol periods but does not assume the knowledge of the actual channel coefficients.

An example of the last scenario is the non-coherent demodulation of differential phase shift keying (DPSK), where the phase reference is assumed to be unknown but remains constant over two symbol periods. Generalization of DPSK to more than two symbol periods can be found in Divsalar and Simon [6].

In the following sections, we refer to T as the *coherence time*, although its interpretation depends on which of the above situations one has in mind.

Because of the independence of the different coherence intervals, to calculate channel capacity it is sufficient to study one coherence interval, where each transmit antenna sends a $T-$*dimensional* vector, and each receive antenna receives a $T-$*dimensional* vector. In complex baseband representation, the system can be written as following:

$$\mathbf{Y} = \mathbf{H}\mathbf{X} + \mathbf{W}, \tag{1}$$

where $\mathbf{X} \in \mathcal{C}^{N \times T}$, and the row vectors $\mathbf{x}_i^t,,i = 1 \ldots N$ correspond to the transmitted signal at the i^{th} transmit antenna. Similarly, $\mathbf{Y} \in \mathcal{C}^{N \times T}$, with each row vector $\mathbf{y}_j^t, j = 1 \ldots N$, is the received signal for the j^{th} receive antenna.

The propagation gain from the j^{th} transmit antenna to the i^{th} receive antenna $\mathbf{h}_{ij}, i, j = 1 \ldots N$ are i.i.d. complex Gaussian $\mathcal{CN}(0, 1)$ distributed. The additive noise $\mathbf{W} \in \mathcal{C}^{N \times T}$ has i.i.d. entries $w_{nt} \sim CN(0, \sigma^2)$. We normalize the equation to let the average transmitted power at each transmit antenna in 1 symbol period to be 1, so the power constraint can be written as

$$E[\sum_{i=1}^{N}\sum_{t=1}^{T} x_{it}] = NT. \tag{2}$$

We refer to the SNR ρ as the average signal to noise ratio at each receiver antenna. Under the normalization above, $\rho = N/\sigma^2$.

The capacity of the channel (in bits per unit time) is given by:

$$C(\rho) = \frac{1}{T} \sup_{p_{\mathbf{X}}(\cdot)} I(\mathbf{X}; \mathbf{Y}), \tag{3}$$

where the optimization is over all input distribution of $\mathbf{X}$ satisfying the power constraint (2).

The goal of this paper is to compute high SNR approximations to $C(\rho)$ for various values of N and T. All approximations are in the sense that the *difference* of the approximation and $C(\rho)$ goes to zero as the SNR ρ tends to infinity

2.2 Basic Results

We review several results from [4].

Lemma 1 *For any coherence time T and number of receive antennas, the capacity obtained with $N > T$ transmit antennas can also be obtained by $N = T$ transmit antennas.*

Lemma 2 *The signal matrix that achieves capacity can be written as $\mathbf{X} = \mathbf{A}\Theta$, where Θ is an $N \times T$ isotropically distributed unitary matrix and is independent of $\mathbf{A}$. $\mathbf{A}$ is an $N \times N$ real diagonal matrix such that the joint distribution of the diagonal entries is exchangeable (i.e. invariant to the permutation of the entries).*

A random matrix Θ is called *isotropically distributed* (i.d.) if its distribution is invariant to rotations, ie. given arbitrary unitary matrix $Q \in C^{T\times T}$, $P(\Theta) = P(\Theta Q)$. As an example, we observe that the matrix of channel coefficients $\mathbf{H}$ is i.d. For detailed discussion of i.d. matrices, please refer to [4].

The ith row θ_{x_i} of Θ represents the direction of the transmitted signal from antenna i, and the ith diagonal entry $\|\mathbf{x}_i\|$ represents the norm of that signal. This characterization reduces the dimensionality of the optimization problem from NT to N by specifying the distribution of the signal directions, but observe that the optimal distribution of the norms is not determined. For the rest of the paper, we will without loss of generality consider input distributions within this class.

2.3 Stiefel and Grassmann Manifolds

Natural geometric objects of relevance to the problem are the Stiefel and Grassmann manifolds. The *Stiefel manifold* $S(T,N)$ for $T \geq N$ is defined as the set of all unitary $N \times T$ matrices, i.e.

$$S(T,N) = \{Q \in \mathcal{C}^{N\times T} : QQ^{\dagger} = I_N\}.$$

In the special case of $N = 1$, this is simply the surface of the unit sphere in $\mathcal{C}^T$.

The Stiefel manifold $S(T,N)$ can be viewed as an embedded sub-manifold of $\mathcal{C}^{N\times T}$ of (complex) dimension $2TN - N(N+1)+1$. One can define a measure μ on the Stiefel manifold, called the *Haar measure*, induced by the Lebesgue measure on $\Re^{2NT-N(N+1)+1}$ through this embedding. It can be shown that this measure is invariant under rotation, i.e. if $\mathcal{S}$ is a measurable subset of $S(T,N)$, the integral with respect to the Haar measure μ satisfies:

$$\int_{\mathcal{S}} \mu(dQ) = \int_{\mathcal{S}P} \mu(dQ)$$

for any unitary $T \times T$ matrix P. Hence, *an i.d. unitary matrix is uniformly distributed on the Stiefel manifold with respect to the Haar measure.* In the case of $N = 1$, the Haar measure is simply the uniform measure on the surface of the unit sphere.

The total volume of the Stiefel manifold as computed from this measure is given by

$$|S(T,N)| = \int_{S(T,N)} \mu(dQ) = \prod_{i=T-N+1}^{T} \frac{2\pi^i}{(i-1)!}. \tag{4}$$

We can define the following equivalence relation on the Stiefel manifold : two elements $P, Q \in S(T,N)$ are equivalent if the row vectors (T dimension) span the same subspace, i.e. $P = UQ$ for some unitary $N \times N$ matrix U. The *Grassmann manifold* $G(T,N)$is defined as the quotient space of $S(T,N)$ with respect to this equivalence relation. Each element in the Grassmann Manifold $G(T,N)$ is an equivalence class in $S(T,N)$. In other words, $G(T,N)$ is the set of all N dimensional subspaces of $\mathcal{C}^T$. The dimension of the Grassmann manifold is given by

$$\dim(G(T,N)) \;=\; \dim(S(T,N)) - \dim(S(N,N)) = N(T-N).$$

The Haar measure on the Stiefel manifold induces a natural measure on the Grassmann manifold. The resulting volume of the Grassmann manifold is:

$$|G(T,N)| \;=\; \frac{|S(T,N)|}{|S(N,N)|} = \frac{\prod_{i=T-N+1}^{T} \frac{2\pi^i}{(i-1)!}}{\prod_{i=1}^{N} \frac{2\pi^i}{(i-1)!}}.$$

3 The Multi-Antenna Channel

In this section, we will study the multiple antenna fading channel (1). Since both transmitted and received signals are matrices, $\mathbf{X}$ and $\mathbf{Y}$ respectively, so we often need to compute the differential entropy of a random matrix. To do this, a seemingly natural way is to view a $N \times T$ matrix as a vector of length NT, and compute the differential entropy in the rectangular coordinate system in $\mathcal{C}^{N \times T}$. However, the fact that the optimal input $\mathbf{X}$ has isotropic directions Θ suggests to use a different coordinate system. In this section, we will start with computing the capacity by using this new coordinate system, and then give a geometric interpretation of the results.

3.1 Capacity for multiple antenna channel

For convenience, we rewrite the channel model (1) here.

$$\begin{aligned} \mathbf{Y} &= \mathbf{H}\mathbf{X} + \mathbf{W} \\ &= \mathbf{H}\mathbf{A}\Theta + \mathbf{W}, \end{aligned}$$

where $\mathbf{Y}, \mathbf{X}, \mathbf{W}, \Theta \in \mathcal{C}^{N \times T}$, $\mathbf{H}$ is $N \times N$ iid $\mathcal{CN}(0,1)$ matrix, $\mathbf{A} \in \mathcal{R}^{N \times N}$ is diagonal, representing the norms of the input signals on the transmit antenna. The matrix Θ is i.d.

We focus first on the case $T \geq 2N$. We start with the following lemma.

Lemma 3 *Let $\mathbf{x}_i^{(\sigma)}, i = 1..N$ be the optimal input signal for each transmit antenna at noise level σ. If $T \geq 2N$, then*

$$\frac{\sigma}{\|\mathbf{x}_i^{(\sigma)}\|} \xrightarrow{P} 0$$

for $i = 1..N$

This lemma says that to achieve capacity at high SNR, a signal that is much higher than the noise level with high probability must be sent at each transmit antenna. Essentially, this is similar to the situation that in the high SNR regime of the AWGN channel, it is much more preferable to spread the available energy over all degrees of freedom rather than transmit over only a fraction of the degrees of freedom.

Now,

$$I(\mathbf{X}; \mathbf{Y}) = h(\mathbf{Y}) - h(\mathbf{Y} \mid \mathbf{X}).$$

To compute the conditional entropy $h(\mathbf{Y} \mid \mathbf{X})$, notice that given $\mathbf{X} = X$, $\mathbf{Y}$ is Gaussian, so we have:

$$\begin{aligned} h(\mathbf{Y} \mid \mathbf{X}) &= NE[\sum_{i=1}^{N} \log(\pi e(\|\mathbf{x}_i\|^2 + \sigma^2))] + N(T-N)\log \pi e \sigma^2 \\ &\approx NE[\sum_{i=1}^{N} \log(\pi e \|\mathbf{x}_i\|^2)] + N(T-N)\log \pi e \sigma^2. \end{aligned} \tag{5}$$

Lemma 3 can be used to justify that the approximation error in the last step approaches 0 at high SNR. To compute $h(\mathbf{Y})$, we need the following lemma.

Lemma 4 Change of Coordinates
Let $\mathbf{R}$ *be a random matrix in* $\mathcal{C}^{N\times T}$, *with* $N \leq T$, *and let the SVD be* $\mathbf{R} = \mathbf{U}\Sigma\mathbf{V}^{\dagger}$, *where* $\mathbf{U}$ *is* $N\times N$ *unitary matrix,* Σ *is* $N\times N$ *diagnal matrix with non-negative entries,* $\mathbf{V}$ *is* $T\times N$ *unitary matrix. If both* $\mathbf{R}$ *and* $\mathbf{R}^{\dagger}$ *are i.d. , then*

$$h(\mathbf{R}) = h(\mathbf{U}\Sigma\mathbf{Q}) + \log|G(T,N)| + (T-N)E[\log\det(\Sigma)^2] \tag{6}$$

where $\mathbf{Q}$ *is i.d. unitary in* $\mathcal{C}^{N\times N}$ *and independent of* $\mathbf{U}\Sigma$.

Lemma 4 can be viewed as a consequence of a coordinate transformation, which represents $\mathbf{R}$ by the subspace $\Omega_{\mathbf{R}}$ spanned by its row vectors ($\Omega_{\mathbf{R}} \in G(T,N)$) together with a $N\times N$ matrix $\mathbf{C}_{\mathbf{R}}$ which specifies the location of $\mathbf{R}$ with respect to a prescribed basis in $\Omega_{\mathbf{R}}$. Because $\mathbf{R}$ is i.d., irrespective of the choice of the basis, $\mathbf{C}_{\mathbf{R}}$ can be expressed as $\mathbf{U}\Sigma\mathbf{Q}$ with Q i.d. unitary. The key observation is since $\mathbf{R}$ is i.d., thus $\Omega_{\mathbf{R}}$ is uniformly distributed on Grassmann manifold $G(T,N)$, hence we can write $h(\Omega_{\mathbf{R}}) = \log|G(T,N)|$. The last term in formula is the Jacobian of this changing of coordinates.

For i.d. matrix $\mathbf{R}$, the joint distribution of the entries are complicated, so it is hard to compute the differential entropy directly in rectangular coordinates. By introducing the new coordinates above, the problem is greately simplified.

Now we make the approximation $h(\mathbf{Y}) \approx h(\mathbf{HX})$, which can again be justified by Lemma 3. Applying Lemma 4 for $\mathbf{HX} = \mathbf{U}\Sigma\mathbf{V}^{\dagger}$, we get

$$\begin{aligned} h(\mathbf{Y}) &\approx h(\mathbf{HA}\Theta) \\ &= h(\mathbf{U}\Sigma\mathbf{Q}) + \log|G(T,N)| + (T-N)E[\log\det\Sigma^2] \\ &= h(\mathbf{U}\Sigma\mathbf{Q}) + \log|G(T,N)| + (T-N)E[\log\det\mathbf{A}^2] + (T-N)E[\log\det(\mathbf{HH}^{\dagger})] \end{aligned} \tag{7}$$

where $\mathbf{Q}$ is i.d. unitary $N\times N$ matrix independent of $\mathbf{U}$ and Σ. Notice $\mathbf{U}\Sigma\mathbf{Q}$ can be written as $\mathbf{HA}\tilde{\mathbf{Q}}$ with $\tilde{\mathbf{Q}}$ being i.d. unitary $N\times N$ matrix independent of $\mathbf{HA}$. Combining with (5) and (7), we get:

$$\begin{aligned} I(\mathbf{X};\mathbf{Y}) \;=\; & \log|G(T,N)| + \log\det\mathbf{HH}^{\dagger} - N(T-N)\log\pi e\sigma^2 \\ & + h(\mathbf{HA}\tilde{\mathbf{Q}}) + (T-2N)E[\log\det\mathbf{A}^2] - N^2\log\pi e + o(1). \end{aligned} \tag{8}$$

Now,

$$E\left[\sum_{i,j=1}^{N}\left|\left(\mathbf{HA}\tilde{\mathbf{Q}}\right)_{ij}\right|^2\right] = NE\left[\sum_{i=1}^{N}\|\mathbf{x}_i\|^2\right] = N^2 T,$$

so the differential entropy term in (8) is upper bounded by $N^2\log\pi eT$, achieved if the entries of $\mathbf{HA}\tilde{\mathbf{Q}}$ are all iid. $\mathcal{CN}(0,T)$. This is the case when all the norms $\|\mathbf{x}_i\| = \sqrt{T}$ with probability 1. If $T \geq 2N$, the term $(T-2N)E[\log\det\mathbf{A}^2]$ is also maximized by this choice of input distribution due to the concavity of the log function. Thus, the constant norm input distribution maximizes the approximate mutual information (8).

A precise statement of the result is contained in the following Theorem.

Theorem 5 *Assume* $T \geq 2N$*. The high SNR capacity of a system with* N *receive and* N *transmit antennas (bps/Hz) is given by:*

$$\begin{aligned} C(\rho) &= \frac{1}{T}\log|G(T,N)| + \frac{T-N}{T}E[\log\det(\mathbf{H}\mathbf{H}^{\dagger})] \\ &\quad + N(1-\frac{N}{T})\log\frac{T\rho}{N\pi e} + o(1). \end{aligned} \tag{9}$$

where $E[\log\det(\mathbf{H}\mathbf{H}^{\dagger})] = \sum_{i=1}^{N} E[\log\chi_{2i}^2]$ *with* χ_{2i}^2 *a Chi-square random variable of dimension* $2i$*.*

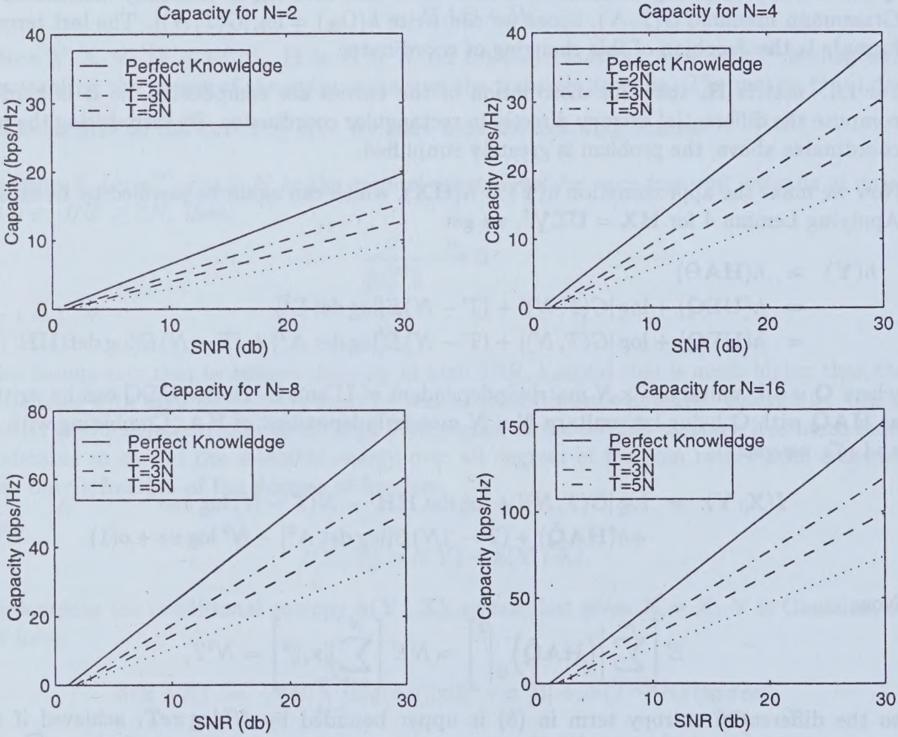


Figure 1: High SNR capacity of multiple antenna channel

Figure 1 shows a few examples of capacity in multiple antenna channel, comparing to the perfect knowledge result in [1]:

$$C = E[\log\det(\mathbf{H}\mathbf{H}^\dagger)] + N\log\frac{\rho}{N} + o(1).$$

We observe that, as expected, the capacity approaches the perfect knowledge case as T increases.

For the case $T < 2N$, we have the following result.

Theorem 6 *If $T < 2N$, the high SNR capacity is of the form:*

$$C(\rho) = C_0 + \frac{1}{T}\lfloor\frac{T}{2}\rfloor\left(T - \lfloor\frac{T}{2}\rfloor\right)\log\rho + o(1).$$

where C_0 is a constant that does not depend on ρ.

In this case, the asymptotically optimal input distribution uses only $\lfloor T/2\rfloor$ of the N transmit antennas. The reason is as follows. When k transmit antennas are used, the number of degrees of freedom available for conveying information using the subspace Ω is $k(T-k)$, the dimension of the Grassmann manifold $G(T,k)$. For a fixed T, this dimension first increases with k, reaches the maximum value at $k^* = \lfloor T/2\rfloor$ and then decreases again. If $T \geq 2N$, i.e. $N \leq T/2$, then this dimension is maximized by using all N transmit antennas. This corresponds to the case analyzed earlier. On the other hand, if $T < 2N$, then this dimension is maximized by using only k^* of the available N transmit antennas. This yields the fastest rate of growth of mutual information with the SNR. However, in this case, the optimal strategy is in general not to have constant input norm for the k^* transmit antennas, and the optimal norm distribution can only be computed numerically.

3.2 Geometric Interpretation

We can represent the transmitted signal $\mathbf{X}$ in terms of the N-dimensional subspace Ω_X spanned by the row vectors of $\mathbf{X}$ and its complex coefficient $\mathbf{C_X} = \mathbf{A}\tilde{\mathbf{Q}}$ with respect to a prescribed basis in Ω_X. Under the constant norm input distribution, the observation $\mathbf{Y}$ provides no information about the phase $\tilde{\mathbf{Q}}$, since the fading gain $\mathbf{H}$ is i.d. Moreover, the subspace is independent of $\mathbf{A}\tilde{\mathbf{Q}}$. This means that under the constant norm input distribution,

$$I(\mathbf{X};\mathbf{Y}) = I(\Omega_X;\mathbf{Y}) = h(\Omega_X) - h(\Omega_X \mid \mathbf{Y}).$$

If the differential entropy is computed with respect to the Haar measure on the Grassman manifold, then

$$h(\Omega_X) = \log|G(T,N)|,$$

since the input is i.d. Hence, from eqn. (9), we get:

$$\begin{aligned} h(\Omega_X \mid \mathbf{Y}) &= h(\Omega_X) - I(\Omega_X;\mathbf{Y}) \\ &= -(T-N)E[\log\det(T\mathbf{H}\mathbf{H}^\dagger)] + N(T-N)\log\pi e\sigma^2 + o(1). \end{aligned}$$

Here we can see the intuitive reason to use isotropic Θ to maximize $h(\Omega)$, and the motivation of using the changing of coordinate system in Lemma 4 to capture the essence of the problem.

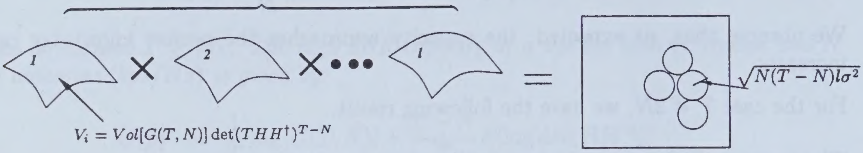


Figure 2: Sphere Packing in the Grassmann Manifold

Analogous to the sphere packing argument in AWGN channel, the result above can be interpreted as packing spheres in the Grassmann manifold. In each coherence interval, we send a point in the Grassmann manifold $G(T,N)$. The volume of the Grassmann manifold $|G(T,N)|$ is scaled by the factor of $det(THH^\dagger)^{T-N}$ as the received power. If we code over l coherence intervals, we just need to select code word from the product space of l Grassmann manifolds of total dimension $N(T-N)l$. as shown in Figure 2. Corrupted by the additive noise, the received signal lies in a sphere of radius $\sqrt{N(T-N)l\sigma^2}$ centered at the transmitted code word with high probability for large l. The number of reliably distinguishable codewords is the number of these spheres packed, which is given asymptotically by the ratio of the two volumes.

References

[1] G.J.Foschini, "Layered space-time architecture for wireless communication in a fading environment when using multi-element antennas," *Bell Labs Technical Journal*, vol. 1, no. 2, pp. 41–59, 1996.

[2] G. Foschini and M. Gans, "On limits of wireless communications in fading environment when using multiple antennas," *Wireless Personal communications*, 1998.

[3] I. Telatar, "Capacity of multi-antenna gaussian channels," *AT&T Bell Laboratories internal Technical Memorandum*, 1995.

[4] T. Marzetta and B. Hochwald, "Capacity of mobile multiple-antenna communication link in a rayleigh flat-fading environment," *IEEE Transactions on Information Theory*, vol. 45, no. 1, pp. 139–57, 1999.

[5] B. Hochwald and T. Marzetta, "Space-time modulation scheme for unknown rayleigh fading environments," in *Proceedings of 36th Annual Allerton conference.*

[6] D. Divsalar and M. Simon, "Multiple-symbol differential detection for mpsk," *IEEE Transactions on Communications*, vol. 38, no. 3, pp. 300–8, 1990.

Blind Adaptive Multiuser Detection for DS/SSMA Communications with Generalized Random Spreading in a Frequency-Selective Fading Channel*

Joon Ho Cho and James S. Lehnert

School of Electrical and Computer Engineering
Purdue University
West Lafayette, IN 47907-1285
lehnert@purdue.edu

Abstract

In this paper, a generalized pseudo-random spreading format and a decentralized blind adaptive algorithm based on the maximum signal-to-interference power ratio (MSIR) criterion are proposed for direct-sequence spread-spectrum multiple-access (DS/SSMA) communications in a slowly-varying frequency-selective fading channel. Retaining minimal random processing gain and assigning most of the system bandwidth to the signature waveform, an easy estimation of the channel response is possible and significant MAI suppression can also be achieved. Monte-Carlo simulations, as well as a semi-analytical method, show that the proposed adaptive algorithm performs well in frequency-selective fading channels with various delay spreads and/or in near-far situations. With low computational complexity, it estimates the channel response and suppresses both the multiple-access interference and intersymbol interference.

1 Introduction

A major challenge encountered in direct-sequence spread-spectrum multiple-access (DS/SSMA) communications in frequency-selective fading channels is to perform a reliable detection of multiple users' transmitted data based on the received signal which is corrupted by multiple-access interference (MAI) and intersymbol interference (ISI).

A conventional approach to this problem is to use a RAKE receiver [1]. This receiver is decentralized in that it processes the received signal from a particular transmitter without any knowledge of the interfering users' received signals. Unfortunately, the performance is highly limited by the interference [2], so that a centralized RAKE receiver has been proposed [3]. Although the centralized receiver shows improved performance over the decentralized receiver, it requires an accurate channel estimation of all the users and a complicated control for successive interference cancellation.

For Gaussian multiple-access channels, the minimum mean-squared-error (MMSE) approach has been quite successful in MAI suppression. It has optimum near-far resistance and allows simple adaptive and decentralized implementation when periodic spreading sequences are employed [4]. This receiver requires knowledge only of the timing and spreading waveform of the desired user's signal for adaptation, so that in order to apply it to the frequency-selective multiple-access channel we need to know only the timing and channel response of the desired user's signal. Although a simple approach that employs a mismatched spreading waveform may work acceptably in some cases,

---

*The research described in this paper is supported by the US Government DARPA GloMo Project AO No. F383, AFRL contract number F30602-97-C-0314.

this results in a large performance degradation when the delay spread of the channel is not negligible. Furthermore, this approach does not perform an estimation of the spatio-temporal channel response that is required for effective MMSE interference suppression using antenna arrays. Actually, the channel estimation problem can be solved by transmitting a known data sequence to train the receiver. However, whenever there is an abrupt change in the channel environment, the receiver may require a new training sequence which terminates the continuous data transmission.

For *blind* adaptive channel estimation and MMSE detection in a slowly-varying frequency-selective fading channel, subspace approaches have been proposed [6], [7]. Although improved performance is obtained without training sequences, the computational complexity may be prohibitive since the techniques rely on the spectral decomposition of the signal correlation matrix.

Recently, Wong *et. al* proposed a spatio-temporal matched filter receiver for systems with aperiodic random spreading sequences in a slowly-varying frequency-selective fading channel [8]. This receiver is similar to the MAI suppression receivers with symbol-time-period spreading sequences in that it is decentralized and that it suppresses the MAI based on the MMSE, equivalently MSIR (maximum signal-to-interference power ratio), criterion by exploiting the cyclostationarity of the interference components. In general, the chip waveform which results in the cyclostationarity of the interfering signal has small excess bandwidth, and the antenna elements are usually outnumbered by the interfering users. Hence, the MAI suppression capability of this receiver is significant only for the cases with a small number of strong interfering users [10]. This receiver, however, has a great advantage over the systems with periodic spreading sequences in that the spatio-temporal channel response can be easily estimated from the two statistics containing the interfering-signal-only and the desired-plus-interfering-signal components by exploiting knowledge of the known spreading sequence of the desired user.

Motivated by these observations, we propose a generalized random spreading and an accompanying decentralized adaptive algorithm for blind adaptive multiuser detection in a slowly-varying frequency-selective fading channel. The generalized spreading format provides a framework to understand spreading as a way to allocate the large system bandwidth to the random processing gain and the deterministic signature waveform. The blind adaptive algorithm exploits knowledge of the known spreading sequence of the desired user, and easily estimates the spatio-temporal channel response of the desired user's signal.

The organization of this paper is as follows. In Section 2, the signal and system model is described, and the generalized random spreading is proposed. The output statistic is analyzed and the blind adaptive algorithm is proposed in Section 3. Numerical results are given in Section 4, and concluding remarks are offered in Section 5.

2 Signal and System Model

There are K active users. The complex baseband equivalent of the kth ($1 \leq k \leq K$) DS/SS signal $s_k(t)$ is given by

$$s_k(t) = \sum_{n=-\infty}^{\infty} b_k[\lfloor n/N_a \rfloor]\, a_k[n]\, \phi_k\left(\frac{t - nT_p}{T_c}\right), \tag{1}$$

where $\{b_k[m] \in \{-1,+1\}\}_{m=-\infty}^{\infty}$ and $\{a_k[n] \in \{+1,+j,-1,-j\}\}_{n=-\infty}^{\infty}$ are the biphase data bit sequence and long-period quadriphase pseudo-random sequence of the kth user.

The signature waveform $\phi_k(\tau)$ of the kth user is given by

$$\phi_k(\tau) = \sum_{n'=1}^{N_p} c_k[n'] \, \psi(\tau - n'), \tag{2}$$

where $\{c_k[n'] \in \{+1, +j, -1, -j\}\}_{n'=1}^{N_p}$ is the quadriphase signature sequence of the kth user and $\psi(\tau)$ is the chip waveform which is a real function and normalized, i.e., $\int_{-\infty}^{\infty} \psi(\tau)^2 \, d\tau = 1$. $T_b \triangleq N_a T_p$ is the bit duration, T_p is the signature waveform duration, and $T_c \triangleq T_p / N_p$ is the chip duration. The parameter N, defined by $N \triangleq N_a N_p$, is called the processing gain which indicates the number of chips assigned to a bit transmission. We call N_a the random processing gain.

The signal format presented in (1) and (2) includes the most commonly used spreading schemes as two extremes. First, consider the case with $N = N_a$. In this case, the spreading sequences become pure pseudo-random sequences, which are modeled as aperiodic random sequences. Usually, the large system bandwidth is consumed in increasing the random processing gain, and consequently the chip waveform has very small excess bandwidth. Therefore, the MAI mitigation capability mainly comes from the *random* spreading gain and slightly from the chip waveform when the conventional matched filter receiver is employed to demodulate the signal. The random processing gain provides a robust MAI suppression in that the effect of an interfering user's signal in the decision statistic is suppressed by the factor of N_a, whatever power and delay profile it has. Second, consider the case with $N = N_p$, i.e., $N_a = 1$. In this case, the spreading sequences become pure periodic sequences with a symbol-time period. A lot of linear receivers have been proposed for this spreading scheme. One of the most important parameters is the signal space dimension, which is determined by N_p. The significant MAI mitigation capability of these receivers is obtained mainly by exploiting the cyclostationarity of the MAI components with a symbol-time period as well as the large excess bandwidth of the (*deterministic*) signature waveforms. This linear suppression is efficient in that the effect of an interfering user's signal in the decision statistic can be suppressed more than by a factor of N_a under some conditions.

Besides these two spreading schemes, we can also consider the cases with $1 < N_a < N$. For these cases, we compromise between robustness and efficiency by properly selecting N_a and N_p, whose product is N, a parameter that is closely related to the excess bandwidth of the data symbol waveform.

In this way, the generalized pseudo-random spreading format provides a unified framework to understand DS/SS spreading as a way to allocate large system bandwidth to the random processing gain and to the excess bandwidth of a deterministic waveform. In this paper, we mainly consider the cases with small $N_a (\geq 2)$. Hence, we develop a receiver that suppresses the MAI mainly by exploiting the cyclostationarity and the large excess bandwidth of the signature waveforms, rather than by the random spreading.

The signal $s_k(t)$ is modulated by a carrier and transmitted through a slowly-varying frequency-selective multipath fading channel. If an asynchronous uplink is considered, then the complex baseband equivalent of the received signal can be written as

$$r(t) = \sum_{k=1}^{K} \sum_{l=1}^{L_k} \alpha_{k,l} s_k(t - T_{k,l}) + u(t), \tag{3}$$

where L_k is the total number of multipath components from the kth mobile user, and $\alpha_{k,l}$ is the complex fading coefficient of the multipath component corresponding to the propagation delay $T_{k,l}$. Finally, $u(t)$ is the complex additive white Gaussian noise (AWGN) process with two-sided power spectral density N_o.

We assume $T_{k,1} < T_{k,2} < \cdots < T_{k,L_k}$ for $1 \leq k \leq K$, without loss of generality. In addition, we also assume that the delay spread is less than one-N_ath of the bit duration, so that it satisfies $\lfloor \tau_{k,L_k} \rfloor - \lfloor \tau_{k,1} \rfloor < N/N_a = N_p$ for all k, where $\tau_{k,l} \triangleq T_{k,l}/T_c$ is the normalized delay of $T_{k,l}$.

Consider the detection of the first user. We assume that a synchronization subsystem, which synchronizes the pseudo-random sequence generators at the transmitter and receiver, has already performed a timing recovery of the desired user's signal. The receiver processes the output samples of the chip matched filter and the samples are taken at ain nteger multiple of T_c.

We define the N-by-1 vector $\mathbf{r}[m]$, consisting of the N samples of the chip matched filter output corresponding to the mth bit, as $\mathbf{r}[m] = \left[\ \mathbf{r}^{(1)}[m]^T, \mathbf{r}^{(2)}[m]^T, \cdots, \mathbf{r}^{(N_a)T}[m] \ \right]^T$, where the samples are ordered by the sampling time and the $\mathbf{r}^{(\cdot)}[m]$'s are N_p-by-1 vectors. We consider a linear receiver. Hence, the decision statistic $z[m]$ is given as

$$z[m] = \mathbf{w}[m]^H \left\{ \sum_{n=1}^{N_a} a_1[N_a m + n - 1]^* \mathbf{r}^{(n)}[m] \right\} \triangleq \mathbf{w}[m]^H \mathbf{z}[m].$$

Since the proposed algorithm does not recover the desired user's phase information, in order to perform coherent detection we need a phase estimation subsystem. In this paper, we do not consider any specific phase recovery algorithm. Instead, we employ simple differentially coherent detection as

$$\hat{d}_1[m] = \operatorname{sign}\left\{ \operatorname{Re}(z[m] z[m-1]^*) \right\}, \tag{4}$$

where $\hat{d}_1[m]$ is the estimate of the original data bit $d_1[m] \in \{1, -1\}$, which is differentially encoded as $b_1[m] = b_1[m-1] d_1[m]$.

3 Output Statistics and Blind Adaptive Algorithm

3.1 Output Statistics

In this subsection, we analyze the output statistic $\mathbf{z}[m]$ given in (4), and an auxiliary output statistic $\hat{\mathbf{z}}[m]$ defined as

$$\hat{\mathbf{z}}[m] = \sum_{n=1}^{N_a} \hat{a}_1[N_a m + n - 1]^* \mathbf{r}^{(n)}[m], \tag{5}$$

where

$$\{\hat{a}_1[N_a m + n - 1] \in \{+1, +j, -1, -j\}\}_{n=1}^{N_a} \tag{6}$$

is the mth auxiliary despreading sequence, which is orthogonal to the mth despreading sequence, i.e.,

$$\sum_{n=1}^{N_a} \hat{a}_1[N_a m + n - 1]^* a_1[N_a m + n - 1] = 0. \tag{7}$$

Since the receiver has the information about the despreading sequences, the auxiliary sequences can be easily found for even N_a. Although there is no orthogonal sequence satisfying both (6) and (7) for odd N_a, we may use auxiliary sequences which are quasi-orthogonal to the corresponding despreading sequences to result in the same, very low magnitude of cross-correlation (e.g., a unit value).

Consider the detection of the 0th bit of the first user. From now on, we only consider the case with $N_a = 2$ and the time-limited chip waveforms defined on $\tau \in [0, 1)$. We assume $-T_c < T_{1,l} < T_c$ for some $1 \leq l \leq L_1$. For mathematical tractability, we model the pseudo-random sequences as aperiodic random quadriphase sequences. Then, for $2 \leq k \leq K$, the effect of the data bit sequence is absorbed in the randomness of the spreading sequence, so that we can set $b_k[m] = 1$ and assume $0 \leq T_{k,1} < T_b/2$, without loss of generality.

The output of the chip matched filter is sampled at $t = 0, T_c, ..., (N-1)T_c$ in order to construct $\mathbf{r}[0]$. To proceed, we define the acyclic down-shift operator $\mathbf{D}_n^l \mathbf{p}$ of a vector $\mathbf{p}$ as follows:

- $\mathbf{p}$ is an n'-by-1 column vector, where $n' > 0$ is an arbitrary integer.
- $\mathbf{D}_n^l \mathbf{p}$ is an n-by-1 column vector.
- If $l \geq 0$, then $\mathbf{D}_n^l \mathbf{p}$ acyclically down-shifts the vector $\mathbf{p}$ l times and then, if necessary, truncates the tail or zero-pads to make it an n-by-1 column vector.
- If $l < 0$, then $\mathbf{D}_n^l \mathbf{p}$ acyclically up-shifts the vector $\mathbf{p}$ $(-l)$ times, truncates the head, and, if necessary, zero-pads to make it an n-by-1 column vector.

By using this definition, we can write $\mathbf{z}[0]$, after proper scaling, as

$$\begin{aligned}
\mathbf{z}[0] &= 2b_1[0]\mathbf{D}_{N/2}^{\lfloor \tau_{1,1} \rfloor}\mathbf{p}_1 \\
&+ \begin{pmatrix} b_1[-1]a_1[-1]a_1[0]^* \\ + b_1[0]a_1[0]a_1[1]^* \end{pmatrix} \mathbf{D}_{N/2}^{(\lfloor \tau_{1,1} \rfloor - N/2)}\mathbf{p}_1 + \begin{pmatrix} b_1[0]a_1[1]a_1[0]^* \\ + b_1[1]a_1[2]a_1[1]^* \end{pmatrix} \mathbf{D}_{N/2}^{(\lfloor \tau_{1,1} \rfloor + N/2)}\mathbf{p}_1 \\
&+ \sum_{k=2}^{K} \sum_{l'=-2}^{0} \begin{pmatrix} a_k[l']a_1[0]^* \\ + a_k[l'+1]a_1[1]^* \end{pmatrix} \mathbf{D}_{N/2}^{\left(\lfloor \tau_{k,1} \rfloor + (N/2)l'\right)}\mathbf{p}_k + \mathbf{u}[0],
\end{aligned}$$

where $\mathbf{u}[0]$ is the complex Gaussian random vector with mean zero and covariance $2N_o\mathbf{I}$, and $\mathbf{p}_k$ is the sampled version of the kth user's received signal, given as

$$\mathbf{p}_k = \sum_{l=1}^{L_k} \sum_{l'=0}^{1} \alpha_{k,l}\hat{\psi}\left(\mathrm{mod}(\tau_{k,l}) - l'\right) \mathbf{D}_N^{\left(\lfloor \tau_{k,l} \rfloor - \lfloor \tau_{k,1} \rfloor + l'\right)} \mathbf{c}_k, \tag{8}$$

where $\hat{\psi}(\tau) \triangleq \psi(\tau) * \psi(-\tau)$ and $\mathbf{c}_k \triangleq [c_k[1], c_k[2], \cdots, c_k[N_p]]^T$. Note that the receiver observes different signature waveforms from the transmitted ones.

From (8), the conditional mean and covariance of $\mathbf{z}[0]$ given $b_1[0]$ can be found as

$$E\{\,\mathbf{z}[0] \mid b_1[0]\,\} = 2b_1[0]\mathbf{D}_{N/2}^{\lfloor \tau_{1,1} \rfloor}\mathbf{p}_1 \triangleq b_1[0]\mathbf{z}_1 \ \text{ and} \tag{9}$$

$$\begin{aligned}
&Cov\{\,\mathbf{z}[0] \mid b_1[0]\,\} \\
&= 2\Big\{ \left(\mathbf{D}_{N/2}^{(\lfloor \tau_{1,1} \rfloor - N/2)}\mathbf{p}_1\right)\left(\mathbf{D}_{N/2}^{(\lfloor \tau_{1,1} \rfloor - N/2)}\mathbf{p}_1\right)^H + \left(\mathbf{D}_{N/2}^{(\lfloor \tau_{1,1} \rfloor + N/2)}\mathbf{p}_1\right)\left(\mathbf{D}_{N/2}^{(\lfloor \tau_{1,1} \rfloor + N/2)}\mathbf{p}_1\right)^H \\
&+ \sum_{k=2}^{K} \sum_{l'=-2}^{0} \left(\mathbf{D}_{N/2}^{\left(\lfloor \tau_{k,1} \rfloor + (N/2)l'\right)}\mathbf{p}_k\right)\left(\mathbf{D}_{N/2}^{\left(\lfloor \tau_{k,1} \rfloor + (N/2)l'\right)}\mathbf{p}_k\right)^H + N_o\mathbf{I}\Big\} \triangleq \mathbf{R}_{\hat{\mathbf{z}}}.
\end{aligned} \tag{10}$$

In the same way, we can compute the conditional mean and covariance of $\hat{\mathbf{z}}[0]$ given $b_1[0]$ as

$$E\{\,\hat{\mathbf{z}}[0]\,|\,b_1[0]\,\} = 0 \quad \text{and} \quad Cov\{\,\hat{\mathbf{z}}[0]\,|\,b_1[0]\,\} = \mathbf{R}_{\hat{\mathbf{z}}}. \tag{11}$$

Since $E\{\mathbf{z}[0]\mathbf{z}[0]^H\} = \mathbf{z}_1\mathbf{z}_1^H + \mathbf{R}_{\hat{\mathbf{z}}}(\triangleq \mathbf{R}_{\mathbf{z}})$ and $E\{\hat{\mathbf{z}}[0]\hat{\mathbf{z}}[0]^H\} = \mathbf{R}_{\hat{\mathbf{z}}}$, we have

$$E\{\mathbf{z}[0]\mathbf{z}[0]^H\} - E\{\hat{\mathbf{z}}[0]\hat{\mathbf{z}}[0]^H\} = \mathbf{z}_1\mathbf{z}_1^H, \tag{12}$$

which plays an important role in facilitating the channel response estimation.

3.2 Linear MSIR Detector

In this subsection, we introduce a linear MSIR detector. We define the SIR given a weight vector $\mathbf{w}$ as

$$\mathrm{SIR}(\mathbf{w}) \;\triangleq\; \frac{|E\{\mathbf{w}^H\mathbf{z}[m]\}|^2}{Var\{\mathbf{w}^H\mathbf{z}[m]\}} = \frac{|\mathbf{w}^H\mathbf{z}_1|^2}{\mathbf{w}^H\mathbf{R}_{\hat{\mathbf{z}}}\mathbf{w}}. \tag{13}$$

Then, by applying the Lagrange multiplier theorem, we can find an optimum solution which maximizes the SIR as $\mathbf{w}_{opt} = \alpha\,\mathbf{R}_{\hat{\mathbf{z}}}^{-1}\mathbf{z}_1$, where α is an arbitrary non-zero constant. By substituting $\mathbf{w}_{opt} = \alpha\,\mathbf{R}_{\hat{\mathbf{z}}}^{-1}\mathbf{z}_1$, into (13), we obtain the maximum SIR as $\mathrm{MSIR} = \mathbf{z}_1^H\mathbf{R}_{\hat{\mathbf{z}}}^{-1}\mathbf{z}_1$.

We define the multiuser efficiency (ME), and redefine the asymptotic multiuser efficiency (AME) [11] and the near-far resistance (NFR) [12], as

$$\xi_1(N_o) \triangleq \frac{\mathrm{SIR}_1(N_o)}{E_{b_1}/N_o}, \quad \xi_1 \triangleq \lim_{N_o\to 0}\xi_1(N_o), \;\text{and}\quad \bar{\xi}_1 \triangleq \inf_{\substack{E_{b_k}\ge 0\\ k\neq 1}} \xi_1, \tag{14}$$

where $\mathrm{SIR}_1(N_o)$ is the SIR of the first user at noise density N_o. The ME represents the ratio of the SIR to the single user signal-to-noise (AWGN only) power ratio (SNR), and it is a useful measure of performance when we apply a Gaussian approximation of the interference to analyze the receiver performance.

A necessary condition for nonzero near-far resistance is that the dimension of the interference subspace is strictly less than that of the signal space. However, we should note that even when the MSIR receiver has zero near-far resistance, a reliable communication may be still possible if there exist only a few strong interferers. Especially in the scenario considered in this paper, since we assume a slowly varying channel, a mild power control can be applied, so that the situation where strong interfering signals span every signal space is highly improbable. Hence, the MSIR receiver will still work properly to suppress the interferences.

A similar necessary condition can be found in [7] for the applicability of the subspace method, one of the most successful techniques for blind adaptive multiuser detection. However, this condition may be easily violated if there exist large numbers of MAI and ISI components. Then, the subspace method applying eigen analysis on the estimate of $\mathbf{R}_{\mathbf{z}}$ may not produce a reliable estimation of the desired component because the noise subspace has zero dimension.

For the systems employing the generalized random spreading scheme with $N_a > 1$, we can exploit the property shown in (12) to estimate the desired component even if the noise subspace has zero dimension. Based on this observation, in the next subsection we develop a blind adaptive algorithm which is more efficient than the subspace methods.

3.3 Blind Adaptive Algorithm

In this subsection, we present a blind adaptive algorithm to find the optimum weight vector. The algorithm is based on the deterministic algorithm developed in [8] and is summarized in Table 1.

4 Numerical Results

In this section, we present some numerical results to illustrate the performance of the blind adaptive algorithm. All of the following results assume that $N_a = 2$, that the half-sine chip waveform is used, and that the spreading sequences are selected randomly. We first illustrate the performance in Gaussian asynchronous multiple-access channels. Two cases of the system load are simulated for perfect power control: $K = 10$ (thin lines) and $K = 15$ (thick lines), both with $N = 32$. Fig. 1 shows the simulation results. As expected, the performance of the matched filter receiver is MAI limited, and it performs poorly even in the high SNR region, whereas the blind adaptive MSIR receiver successfully suppresses the MAI.

Fig. 2 illustrates the performance in frequency-selective fading channels. We employ the tapped-delay-line model [1], where the complex channel coefficient $\alpha_{k,l}$'s are generated according to the zero-mean, complex Gaussian distribution with the exponential multipath intensity profile (MIP). Hence, each $\alpha_{k,l}$ for all k and l is independent and the variance is given as $E\{|\alpha_{k,l}|^2\} \propto e^{-\frac{1}{L}(l-1)}$, $l = 1, 2, 3, ..., L_p$. A slowly varying channel is assumed, so that we use a fixed multipath profile for each run and scale the profile properly according to the SNR. Again, an equal-power 10-user system (thin lines) and a 15-user system (thick lines) are considered. The channel parameters are $L = 5$ and $L_p = 10$. The mismatched MSIR receiver, which is matched to the nominal signature waveform, has poor performance, while the blind adaptive receiver performs well suppressing both the MAI and ISI.

The performance in frequency-selective channels is illustrated for different mean delay spreads in Fig. 3. For convenience, L is used to characterize the mean delay spread. The SNR is 15 dB, the processing gain N is 64, and there are $K = 30$ equal-power users. The parameter L_p is 31, and other parameters are the same as before. The figure shows that the MSIR detector performs well, while the performances of the matched filter receiver and the mismatched MSIR receiver degrade as the mean delay spread increases.

Fig. 4 shows the performance in a dynamic multiple-access channel where interferers may enter or exit the channel [14]. The SNR is 20 dB, and $L_p = 1$. At $t = 0$, six interfering users of 10 dB signal power are in the channel. At $t = 2000$, an interfering user of 20 dB signal power enters the channel. At $t = 4000$, the 20 dB and three of the 10 dB interfering users exit the channel. The figure shows that the adaptive algorithm successfully adapts to the channel variation, and suppresses the strong user signal.

The BER performance in a near-far situation is plotted in Fig. 5. The processing gain N is 32, there are five interfering users of 10 dB signal power, $L = 5$ and $L_p = 10$. Although the blind adaptive algorithm performs successfully in this case, it is observed that, in severe near-far situations, the algorithm takes a long time to adapt, or even fails. As mentioned before, a mild power control may be possible in a slowly varying channel. In this case, by roughly synchronizing the power control timings of all the users, we may have a dynamic multiple-access channel with imperfect power control, as considered in Fig. 4, where the proposed algorithm successfully demodulates the signal.

The extension of the proposed receiver for systems with multiple antenna elements is straightforward. A simulation result is illustrated in Fig. 6, where the processing gain

Table 1: Summary of Blind Adaptive Algorithm

---

1) Update the correlation matrices $\mathbf{R}_{\mathbf{z}}[m]$ and $\mathbf{R}_{\hat{\mathbf{z}}}[m]$ as

$$\mathbf{R}_{\mathbf{z}}[m] = \gamma \mathbf{R}_{\mathbf{z}}[m-1] + (1-\gamma)\mathbf{z}[m]\mathbf{z}[m]^H,$$
$$\mathbf{R}_{\hat{\mathbf{z}}}[m] = \gamma \mathbf{R}_{\hat{\mathbf{z}}}[m-1] + (1-\gamma)\hat{\mathbf{z}}[m]\hat{\mathbf{z}}[m]^H,$$

where $0 \leq \gamma < 1$ is the exponential weighting factor [13].
2) Solve the following linear equation for $\mathbf{y}$:

$$\left(\frac{\mathbf{R}_{\mathbf{z}}[m] - \mathbf{R}_{\hat{\mathbf{z}}}[m]}{\mathrm{tr}(\mathbf{R}_{\mathbf{z}}[m] - \mathbf{R}_{\hat{\mathbf{z}}}[m])} - \mathbf{I} \right) \mathbf{y} = \tilde{\mathbf{z}}_1[m, \tilde{m}-1],$$

where $\tilde{\mathbf{z}}_1[m, 0] = \tilde{\mathbf{z}}_1[m-1, \tilde{M}]$, $1 \leq \tilde{m} \leq \tilde{M}$, and

$$\tilde{\mathbf{z}}_1[m, \tilde{m}] = \frac{\mathbf{y}}{||\mathbf{y}||}.$$

3) Update the weight vector, for $1 \leq m' \leq M$, as

$$\mathbf{w}[m, m'] \;=\; c[m, m'] \left\{ \mathbf{I} - \mu \left(\mathbf{I} - \mathbf{P}_{\mathbf{z}}[m] \right) \mathbf{R}_{\hat{\mathbf{z}}}[m] \right\} \mathbf{w}[m, m'-1],$$

where $\mathbf{w}[m, 0] = \mathbf{w}[m-1, M]$, $\mu = 2/\mathrm{tr}(\mathbf{R}_{\hat{\mathbf{z}}}[m])$, and $c[m, m']$ is chosen to make the l'th element of $\mathbf{w}[m, m']$ real positive. The estimate of the projection matrix is given by

$$\mathbf{P}_{\mathbf{z}}[m] = \frac{\tilde{\mathbf{z}}_1[m, \tilde{M}]\tilde{\mathbf{z}}_1[m, \tilde{M}]^H}{\tilde{\mathbf{z}}_1[m, \tilde{M}]^H \tilde{\mathbf{z}}_1[m, \tilde{M}]}.$$

4) Use $\mathbf{w}[m, M]^H \mathbf{z}[m]$ for detection.
5) $m := m + 1$ and goto 1).

N is 16 and the number of equal-power users satisfies $5 \leq K \leq 25$ and the number of antenna elements satisfies $1 \leq G \leq 4$. The SNR is 9 dB. The channel parameters used are $L = 5$ and $L_p = 7$. A uniform linear array with an element spacing of one half wavelength is employed. Each element is a directional antenna with uniform gain for the angles of arrival between −60 and 60 degrees. The figure shows that the proposed algorithm successfully estimates the spatio-temporal channel response and suppresses the MAI and ISI.

Note that the BER performance of the MSIR receiver evaluated by the Gaussian approximation is quite accurate according to the simulations. It is defined as

$$P_{b_1}^{GA}(N_o) = E\left\{ \frac{1}{2} e^{-\xi_1(N_o)\frac{E_{b_1}}{N_o}} \right\}, \tag{15}$$

where the expectation is over the desired user's multiuser efficiency $\xi_1(N_o)$ defined in (14). Since (15) reduces, in an extreme case with $N = N_a$, to the well-known Conditional Gaussian Approximation (CGA) [15] for the systems with purely random spreading sequences, this approximation may be more accurate in our case with $N_a > 1$ than in the case with $N_a = 1$. In the figures presented, the arithmetic average has replaced the statistical average.

5 Conclusions

In this paper, we consider blind adaptive multiuser detection for DS/SSMA communications in a frequency-selective fading channel. We propose the generalized random spreading format and an accompanying decentralized blind adaptive multiuser detector based on an MSIR algorithm. Using the special structure of the proposed spreading format, with low computational complexity, the receiver estimates the channel response of the desired user's signal and suppresses both the MAI and ISI.

References

[1] J. G. Proakis, *Digital Communications*, 3rd Ed., McGraw Hill, 1995.

[2] J. S. Lehnert and M. B. Pursley, "Multipath diversity reception of spread-spectrum multiple-access communications," *IEEE Trans. Commun.*, vol. 35, no. 11, pp. 1189-1198, Nov. 1987.

[3] Y.C. Yoon, R Kohno, and J. Imai, "A spread-spectrum multiaccess systems with cochannel interference cancellation for multipath fading channels," *IEEE J. Select. Areas Commun.*, vol. 11, no. 7, pp. 1067–1075, Sep. 1993.

[4] U. Madhow and M. L. Honig, "MMSE interference suppression for direct-sequence spread-spectrum CDMA," *IEEE Trans. Commun.*, vol. 42, no. 12, pp.3178-3188, Dec. 1994.

[5] M. Honig, U. Madhow, and S. Verdú, "Blind Adaptive Multiuser Detection," *IEEE Trans. Commun.*, vol. 46, no. 1, pp. 91–103, Jan. 1998.

[6] M. Torlak and G. Xu, "Blind multiuser channel estimation in asynchronous CDMA systems," *IEEE Trans. Signal Processing*, vol. 45, no. 1, pp. 137–147, Jan. 1997.

[7] X. Wang and H. V. Poor, "Blind equalization and multiuser detection in dispersive CDMA channel," *IEEE Trans. Inform. Theory*, vol. 44, no. 2, pp. 677–690, Mar. 1998.

[8] T. F. Wong, T. M. Lok, J. S. Lehnert, and M. D. Zoltowski, "A linear receiver for direct-sequence spread-spectrm multiple-access systems with antenna arrays and blind adaptation," *IEEE Trans. Inform. Theory*, vol. 44, no. 2, pp. 659–676, Mar. 1998.

[9] T. F. Wong, T. M. Lok, and J. S. Lehnert, "Asynchronous multiple-access interference suppression and chip waveform selection with aperiodic random sequences," *IEEE Trans. Commun.*, vol. 47, no. 1, pp. 103–114, Jan. 1999.

[10] J. H. Cho and J. S. Lehnert, " Adaptive optimization of receiver chip waveforms for DS/SSMA communications with random spreaing sequences," in *Proc. MILCOM '99 Conf.*, will appear.

[11] S. Verdú, "Optimum multiuser asymptotic efficiency," *IEEE Trans. Commun.*, vol. COM-34, no. 9, pp. 890-897, Sep. 1986.

[12] R. Lupas and S. Verdú, "Linear multiuser detectors for synchronous code-division multiple-access channels," *IEEE Trans. Inform. Theory*, vol. 35, no. 1, pp. 123-136, Jan. 1989.

[13] S. Haykin, *Adaptive Filter Theory*, 2nd Ed., Prentice Hall, 1991.

[14] X. Wang and H. V. Poor, "Blind multiuser detection: A subspace approach," *IEEE Trans. Inform. Theory*, vol. 44, no. 2, pp. 677–690, Mar. 1998.

[15] R. K. Morrow, Jr. and J. S. Lehnert, "Packet throughput in slotted ALOHA DS/SSMA radio systems with random signature sequences," *IEEE Trans. Commun.*, vol. 40, no. 7, pp. 1223–1230, July 1992.

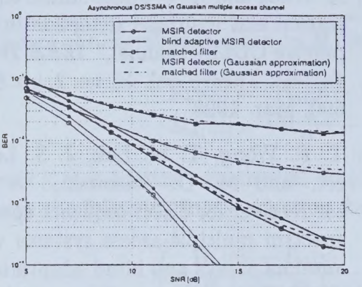


Figure 1: Performance comparison between the conventinal matched filter receiver and the MSIR detector in a Gaussian multiple-access channel.

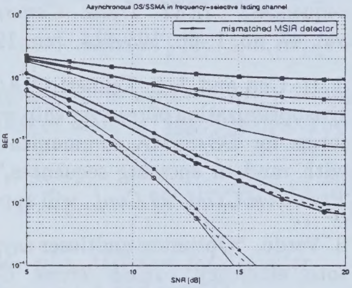


Figure 2: Performance comparison between the conventional matched filter receiver, the mismatched MSIR and the MSIR detector in a frequency-selective fading channel.

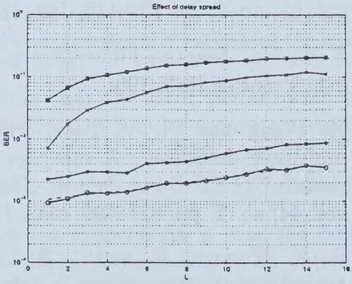


Figure 3: Performance in a frequency-selective channel for various mean delay spreads.

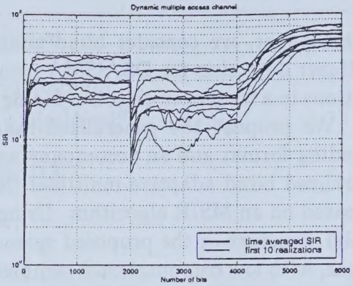


Figure 4: Performance of the blind adaptive MSIR detector in a dynamic multiple-access channel. The first 10 realizations and the time averaged SIR are plotted, where 100 independent runs are averaged.

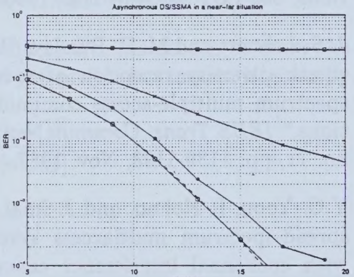


Figure 5: BER performance in a mild near-far situation.

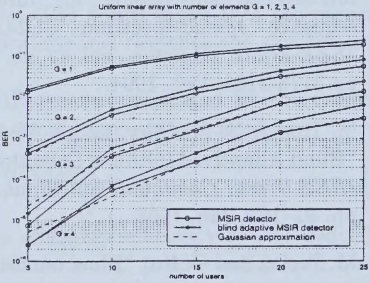


Figure 6: Number of antenna elements and the corresponding increase in system capacity.

Traffic Aided Multiuser Detection for Packet Switching Random Access/CDMA Networks

Biao Chen and Lang Tong[†]
School of Electrical Engineering
Cornell University, Ithaca, NY 14853
Email: biao[ltong]@ee.cornell.edu

Abstract

Traffic burstiness has been observed for packet switching networks due to the multimedia users involved. This burstiness results in the predictability of user activity at the individual source level, and the exploration of such predictability in the receiver design for wireless packet switching random access/CDMA networks is investigated in this paper. Decorrelating detector is considered for performance and complexity tradeoff and it is shown that the conventional approach of assuming all users are active results in substantial performance loss. A two stage multiuser detection is proposed where the first stage tracks active users and feeds its tracking output to the second stage MUD for symbol detection. It is demonstrated that, with the help of traffic predictability, accurate estimate of active set of users is possible, even with a simple matched filter bank implementation at the first stage.

1 Motivation and Introduction

In packet-switching networks, it has been observed that the conventional Poisson modeling is not adequate to capture the traffic burstiness and long range dependence (see, e.g., [4, 7]. Various models have since been proposed in an attempt to better describe the network traffic [3, 11]. In particular, for packet switching networks, recent studies of high-resolution traffic measurement have revealed the so-called self-similar phenomenon that may have potential impact on the design of next generation network.

It is intriguing to notice that self-similarity bears contrary implications for the traffic at the *network* level and the *source* level, the latter referring to the traffic for individual user. On one hand, the direct consequence of self-similarity is the highly variable *network* traffic which is hard to predict as the traffic 'bursts' exist in a wide range of time scale. On the other hand, it has been shown that the cause of self-similarity is the burstiness at the source level, i.e., packets from each user arrive in clusters [12]. *Thus for each individual user, the future activity is actually more predictable as it is highly correlated with the current and previous transmissions.* This predictability is in a sharp contrast to the conventional assumption of Bernoulli traffic which is hard to predict due to the independence assumption among different slots. For broadband wireless networks, such as wireless LAN and wireless ATM, the predictability of individual user's activity could provide valuable information in physical layer designs because of the sharing of a common channel among a number of users.

[†] This work was supported in part by the National Science Foundation under Contract CCR-9804019 and by the Office of Naval Research under Contract N00014-96-1-0895.

The system we consider is a random access/CDMA packet switching network. In particular, we consider the uplink of a slotted packet switching network with DS/SS signaling and with transmitter code, i.e., each user in the network is assigned a different spreading code. Newly arrived packets are encoded with the user's own code and transmitted immediately upon arrival. The system is in line of that considered by Raychaudury [10]. With regard to the receiver, we investigate the implementation of a multiuser detector (MUD), in particular, the decorrelating detector (DD) due to its simplicity, near-far resistance, and analytical tractability.

The implementation of a MUD in a random access network has been studied by a number of authors (see, e.g., [9, 5]). However, a critical issue which has not been given enough attention is that in a random access network, the active user profile is not known to the receiver. Presumably, given that the receiver has the knowledge of all potential users, a MUD could be constructed assuming that all users are active. Such viewpoint however, will incur significant performance loss, which is first observed by Mitra and Poor [6], and will be accurately quantified in this paper. Besides, when the correlation matrix loses rank when the number of users exceeds the spreading gain or simply because of channel distortion, the conventional DD would fail for the users whose signal space is spanned by other users' spreading vectors. Given the performance loss of assuming a 'full model', it is therefore desirable to first infer from the received signal the active set of users that contribute to the current transmission and later build a MUD that uses only the signature waveforms corresponding to the set of active users.

The problem of detecting active users has been addressed by several authors in a dynamic CDMA system [6, 2, 13]. The treatment in [6, 2] differentiates the problem of detection of a new user entering the system and that of an old user ceasing communication. First those old users are examined to see if there is any possible drop-out. For those continuing users, their contribution to the current transmission is zeroed out either by projection or simply by subtracting the estimated signal for those users. The residual signal would contain contributions from a possible new user plus noise. In [13], the detection (identification) of active users is solved using MUSIC. An accurate estimate of correlation matrix as well as the dimension of signal space is required for the ensuing eigen decomposition and identification process.

A common feature of all above approaches is that the detection is focused on a particular data window without accounting for the continuity of the channel traffic. In this paper, we intend to study the possibility of utilizing traffic information in the detection of active users. First, the problem is shown to be equivalent to model selection in a linear regression problem. A set of indicator functions is then incorporated in the regression model to accommodate all possible combinations of active users. This treatment also enables us to fully utilize the traffic predictability at the source level.

The organization of the paper is as follows. The signal model considered in this paper is given in the next section. In section 3, we quantify the effect of including non active user in a decorrelator detector. A two stage MUD is proposed in section 4, whose first stage aims to detect the set of active users. This is followed by some simulation result in section 5. Conclusions are included in section 6.

2 Signal Model

Consider a slotted random access packet switching system with M subscribers, as in [10]. Each user is assigned a unique spreading code and transmits synchronously over a common channel. Assume BPSK for simplicity, the received signal within a packet slot could be written as

$$\mathbf{y}_n(t) = \sum_{k\in\mathcal{I}_n} A^{(k)} b_n^{(k)}(t)\mathbf{c}^{(k)} + \mathbf{z}_n(t), \qquad t = 1,2,\ldots,T \tag{1}$$

where n is the packet index, t is the symbol index within a packet, T is the packet size, k is the user index, $A^{(k)}, \mathbf{c}^{(k)}$ are the amplitude and the signature waveform of user k present at the receiver, $b_n^{(k)}(t)$ is the t^{th} symbol within the n^{th} packet for user k, $\mathbf{z}_n(t)$ is additive white Gaussian noise with zero mean and covariance matrix $\sigma^2\mathbf{I}$, and $\mathcal{I}_n$ is the index set of the active users at slot n, thus $\mathcal{I}_n \subset \{1,2,\ldots,M\}$. The receiver's task is to detect the symbols $b_n^{(k)}$ for all $k \in \mathcal{I}_n$ without prior knowledge of $\mathcal{I}_n$.

3 Effect of Overmodeling

A linear DD can be viewed as a least square (LS) estimator followed by a sign detector. The multiple access interference (MAI) is compensated via premultiplying the MF output by the inverse of correlation matrix, at the cost of increased noise variance. Thus, if a decorrelator intends to mitigate the interference from those users that are not actually transmitting, its lone effect is to enhance the noise level therefore degrade the receiver performance. We start with a precise characterization of this effect.

3.1 Increased variance on LS solution

Consider model (1). Assuming $A^{(k)}$'s are unknown, we can define $s^{(k)}(t) = A^{(k)}(t)b^{(k)}(t)$ and $s^{(k)}(t)$ can take on arbitrary value. Suppress the subscript n, we have

$$\mathbf{y}(t) = \sum_{k\in\mathcal{I}} s^{(k)}(t)\mathbf{c}^{(k)} + \mathbf{z}(t) = \mathbf{C}\mathbf{s}(t) + \mathbf{z}(t) \tag{2}$$

Given $\mathcal{I} = \{k_1,\ldots,k_m\}$, this is a standard linear regression problem where the regressors are the signature waveforms. The maximum likelihood (ML) estimate of $\mathbf{s}(t)$ is the LS solution, a property dictated by the independent Gaussian assumption of the noise. The LS estimate of $\mathbf{s}(t)$ using perfect knowledge of the traffic is given by

$$\hat{\mathbf{s}}(t) = (\mathbf{C}'\mathbf{C})^{-1}\mathbf{C}'\mathbf{y}(t).$$

The ML estimator above is unbiased and achieves the Cramér-Rao bound with error covariance $cov(\hat{\mathbf{s}}(t)) = \sigma^2(\mathbf{C}'\mathbf{C})^{-1}$. A decorrelator simply adds a sign detector after $\hat{\mathbf{s}}(t)$.

By using non-active users' channels, the decorrelating receiver aims to fit the data $\mathbf{y}(t)$ with the following model :

$$\mathbf{y} = [\mathbf{C}\vdots\mathbf{D}]\begin{bmatrix}\mathbf{s}\\ \mathbf{r}\end{bmatrix} + \mathbf{z} \tag{3}$$

with $\mathbf{D}$ represents spreading codes of those non active users and $\mathbf{r}$ being the 'pseudo' message of these users. The ML estimate of $\mathbf{s}(t)$, together with its covariance matrix are given by

$$\begin{aligned}\bar{\mathbf{s}}(t) &= [\mathbf{I}\vdots\mathbf{0}]([\mathbf{C}\vdots\mathbf{D}]'[\mathbf{C}\vdots\mathbf{D}])^{-1}[\mathbf{C}\vdots\mathbf{D}]'\mathbf{y}(t)\\ cov(\bar{\mathbf{s}}(t)) &= \sigma^2[\mathbf{I}\vdots\mathbf{0}]([\mathbf{C}\vdots\mathbf{D}]'[\mathbf{C}\vdots\mathbf{D}])^{-1}\begin{bmatrix}\mathbf{I}\\ \mathbf{0}\end{bmatrix}\end{aligned}$$

Comparing the two estimators, we obtained the following Lemma.

Lemma 1 *The error covariance of the full model satisfies*

$$cov(\bar{\mathbf{s}}) = [\mathbf{C}'\mathbf{C} - \hat{\mathbf{C}}'\hat{\mathbf{C}}]^{-1}\sigma^2 = [\mathbf{C}'(\mathbf{I} - \mathbf{D}(\mathbf{D}'\mathbf{D})^{-1}\mathbf{D})\mathbf{C}]^{-1}\sigma^2 \geq cov(\hat{\mathbf{s}}) \tag{4}$$

where $\hat{\mathbf{C}}$ *is the projection of* $\mathbf{C}$ *onto the space spanned by* $\mathbf{D}$*, i.e.,*

$$\hat{\mathbf{C}} = \mathbf{D}(\mathbf{D}'\mathbf{D})^{-1}\mathbf{D}'\mathbf{C} \tag{5}$$

The equality holds iff the signal space spanned by active users is orthogonal to the signal space spanned by inactive users, i.e., $\mathbf{C}'\mathbf{D} = 0$

A special case which allows a geometrical viewpoint deserves further attention. Consider only a single user is active with signature waveform $\mathbf{c}$ that spans the signal space for the user. The variance for the ML estimate using the full model is $[\mathbf{c}'(\mathbf{I} - \mathbf{D}(\mathbf{D}'\mathbf{D})^{-1}\mathbf{D}')\mathbf{c}]^{-1}$ while the variance for the estimate using true model is $\sigma^2(\mathbf{c}'\mathbf{c})^{-1}$. This is illustrated in Figure 1. In general, the error covariance of the estimator using the full model is the inverse of the projection error matrix from $\mathbf{C}$ associated with active users onto the column space of $\mathbf{D}$ associated with inactive users. The more correlated between the signal space of the active users and that of the inactive users, the larger the estimation error.

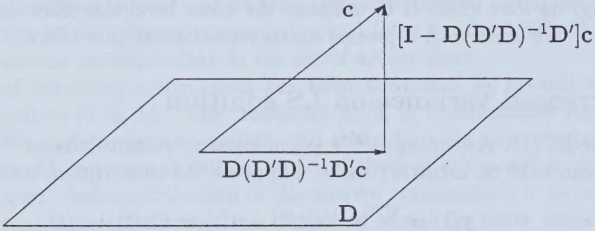


Figure 1: Illustration of the covariance of LS estimate using true and full model. The covariance matrix is "inverse proportional" to the magnitude square of the vector $\mathbf{c}$ for the true model; while for the full model it is "inverse proportional" to the magnitude square of the projection error vector $[\mathbf{I} - \mathbf{D}(\mathbf{D}'\mathbf{D})^{-1}\mathbf{D}']\mathbf{c}$

3.2 Effects on packet success probability

The packet success probability, defined as the average portion of successfully received packets among all packets transmitted, not only depends on the bit error rate (BER) and the error correction code used, but also depends on the channel traffic condition. Conceptually, if the channel is overloaded, employing a full model is not much different than the use of true model. On the other hand, if on average there is only a small number of users being active, then the advantage of correct modeling should be more significant. Assume we have a homogeneous channel traffic, each user at any slot transmitting a packet with probability q, hence the overall number of packets (or equivalently, number of active users, denoted by L) per slot is binomial with success probability q. Define N to be the number of users involved in the DD. Then the packet success probability is

$$P(success) = E_L[P_N(success|L)]$$

where $P_N(success|L)$ is the success probability given L users are active and the decorrelator includes $N \geq L$ users. And for simiplicity we always assume that the active users are included in the decorrelator (no miss detection).

To compute $P_N(success|L)$, assume error correction code can correct up to b bit errors out of a total of T bits (packet length), then

$$P_N(success|L) = \sum_{j=0}^{b} \binom{T}{j} P_e(N)^j (1 - P_e(N))^{T-j}$$

where $P_e(N)$ is the BER when N users are involved in the decorrelator. To allow an explicit expression of $P_e(N)$ as a function of N, assume symmetric spreading vectors with identical pairwise correlation coefficient ρ and perfect power control. It can be shown the BER is

$$P_e(N) = Q\left(\frac{A}{\sigma\sqrt{a(N)}}\right), \quad \text{with} \quad a(N) = \frac{1 + (N-2)\rho}{1 + (N-2)\rho - (N-1)\rho^2}$$

where A is the received amplitude and N is the number of users involved in the decorrelator. For the decorrelator using the full model, $N \equiv M$, hence $P_N(success|L) = P_M(success|L)$ is independent of L. Therefore $P(success)$ is a constant as a function of q. In contrast, for the decorrelator using the true model, $P_N(success|L) = P_L(success|L)$ which results in the dependence of overall packet success probability on the traffic load q. In Figure 2, the packet error rate $(1 - P(success))$ is plotted against q. The gain in terms of packet error rate is of several magnitude over that of the full model. Also plotted in dashed line is the packet error rate using a matched filter (MF) bank for all users. Clearly, as channel load approaches zero, MF receiver should be arbitrarily close to the DD using the true model, as the probability of having more than two active users vanishes.

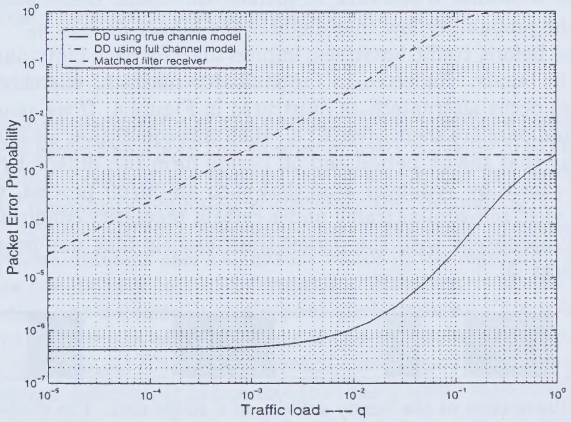


Figure 2: Packet error probability as a function of traffic load parameter q for the decorrelator using full model (dash-dotted line), true model (solid line), and simple matched filter bank (dashed line). The total number of potential users is 30 and the correlation coefficient is assumed to be $\rho = 0.2$ for each pair of signature vectors.

4 Traffic-aided MUD

4.1 Traffic modeling for packet switching networks

The performance degradation of a DD when inactive users' signature waveforms are errantly included provokes the idea of a two stage receiver. In the first stage, the set of active users is detected while in the second stage a MUD is implemented by adapting to the change of user profile, as shown in Figure 3.

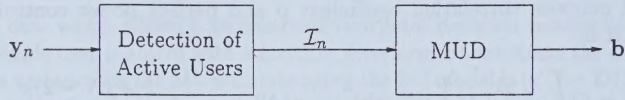


Figure 3: Two-stage multiuser detection

The second stage, the implementation of a MUD is straightforward given the set of active users' signature waveforms. Our focus will be on the first stage. To this end, we first introduce the following model as an alternative to that of (2).

$$\mathbf{y}_n(t) = \sum_{k=1}^{M} \mathbf{c}^{(k)} s_n^{(k)}(t)\gamma_n^{(k)} + \mathbf{z}_n(t)$$

where, in the nth packet slot, $\gamma_n^{(k)}$ is the indicator function that takes value 1 if the k^{th} user is active and 0 otherwise. Incorporating the indicator functions in the signal model not only allows us to accommodate all possible user combinations with a single model, but also provide a natural framework to include the traffic information which will be interpreted as the probability law that dictates the binary process $\gamma_n^{(k)}$.

As mentioned before, traffic burstiness implies that, for each individual user, its transmission occurs in bursts. Reflected in the indicator function, sequences of zeros (idle) and ones (active) occur alternately, as illustrated in Figure 4. If we assume that the duration of zeros and ones are themselves *i.i.d.*, then this constitutes an alternating renewal process (meaning that the on period and off period form two different renewal process). It was actually shown by Willinger *et al* that such an on/off process, when aggregated together, could yield self-similar traffic under certain regularity conditions [12]. Further, such a model is consistent with the packet-train models that was proposed for packet switching networks [3].

Figure 4: Illustration of the bursty activity of a single user. The shaded area indicates the user being actively transmitting.

A first order approximation of the bursty traffic for individual source is a two state Markov chain and was originally proposed by Viterbi in 1986 for the performance analysis of packet switching networks [11]. To see the relationship between the on/off process with the Markov chain approximation, assume that the state transition matrix for $\gamma_n^{(k)}$ is

$$a^{(k)} = \begin{bmatrix} 1-p^{(k)} & p^{(k)} \\ q^{(k)} & 1-q^{(k)} \end{bmatrix}.$$

It is therefore easy to see that conditioned on either state, the duration of staying in that state is geometrically distributed, each with parameter $p^{(k)}$ and $q^{(k)}$. In another word, the two state Markov chain is indeed an alternating renewal process with both on and off duration being geometrically distributed.

4.2 Optimal HMM traffic tracking

Combining all the indicator functions, we define $\Gamma_n = (\gamma_n^{(1)}, \ldots, \gamma_n^{(M)})$ as the state variable of user profile, i.e., those coordinates that equal to '1' in Γ_n indicate that the corresponding users are transmitting at slot n. Given that each individual indicator function is a two state Markov chain, the combined state variable Γ_n is now also a Markov sequence with state dimensionality equal to 2^M where M is the total number of potential users. Its transition matrix $\mathbf{A}$ is simply the Kronecker product of all the transition matrices for each individual $\gamma_n^{(k)}$. In turn, our observation is now a hidden Markov model (HMM).

$$\mathbf{y}_n \triangleq \begin{bmatrix} \mathbf{y}_n(1) \\ \vdots \\ \mathbf{y}_n(T) \end{bmatrix} = \sum_{k \in \mathcal{I}(\Gamma_n)} \begin{bmatrix} \mathbf{c}^{(k)} s_n^{(k)}(1) \\ \vdots \\ \mathbf{c}^{(k)} s_n^{(k)}(T) \end{bmatrix} + \begin{bmatrix} \mathbf{z}_n(1) \\ \vdots \\ \mathbf{z}_n(T) \end{bmatrix} \tag{6}$$

where T is the packet size. Specifically, the underlying state Γ_n evolves according to a Markov transition matrix $\mathbf{A}$, while the actual observation $\mathbf{y}_n$ depends (probabilistically) only on the current state Γ_n. The detection of the set of active users amounts to the inference of the underlying state Γ_n. The optimal state estimator maximizes the posterior probability, $P(\Gamma_n | \mathbf{y}_n, \cdots, \mathbf{y}_1)$, and this could be evaluated using the so-called forward variable, which admits efficient recursive computation [8]. However, the state dimensionality of Γ_n is 2^M, meaning there are total number of 2^M models needs to be evaluated, a usually formidable task in practice.

4.3 Suboptimal traffic tracking

Given the prohibitive complexity of the optimal HMM tracker, a suboptimal approach whose complexity grows linearly in the number of users is highly desirable. We note that the exponential complexity of the optimal HMM tracking is due to the state combining. The key to complexity reduction is therefore the 'decoupling' of individual user's state tracking. Notice that such decoupling inevitably leads to non-optimality as the 'interaction' between different user's signature vector in the regression model is not fully accounted hence results in a loss of information. However, it should be realized that suboptimality does not necessarily imply poor performance. This is because the SNR requirement for the detection of the *presence* of one particular user is much less stringent compared with the required SNR for its symbol detection. Therefore even with a simple front end active-user detector, it is possible that the performance could be close to that of the optimal approach.

Figure 5(a) illustrates a suboptimal traffic tracking algorithm. Instead of expanding all users' states into a single state space, the state tracking is achieved independently for each user hence avoiding the complexity due to the state expansion. Specifically, the received signal is first mapped to a new statistic which serves as the 'likelihood'* of the particular user being active. Then, the status of the user's previous slot is used to jointly decide whether the user is transmitting at the current slot.

---

*Likelihood is an abuse of term since without jointly accounting for all possible set of signature vectors, the obtained 'likelihood' is not strict in its original sense.

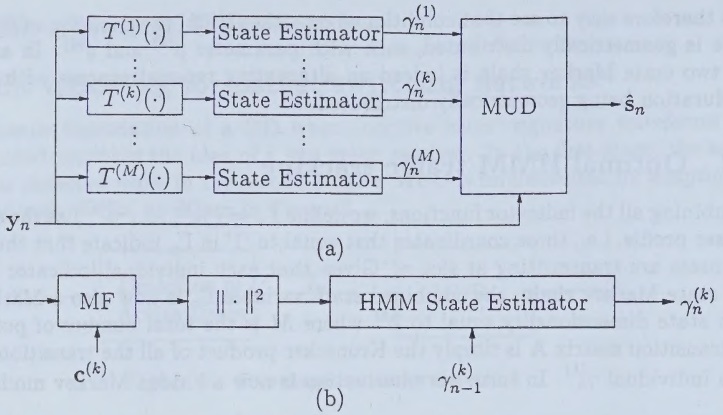


Figure 5: A suboptimal two stage MUD implementation.

Depending on the summary statistic $T^{(k)}(\cdot)$, various structures of user tracker could be formed. In this paper, we propose a simple matched filter front end as shown in Figure 5(b). Note at the first stage, the purpose is to detect the presence of individual users, not the symbol detection, therefore the state tracker should favor simple and robust front end realization. Using Gaussian approximation for MAI, the MF output could be represented as

$$w^{(k)}(t) = s_n^{(k)}(t)\gamma_n^{(k)} + \hat{z}^{(k)}(t), \qquad t = 1, 2, \ldots, T$$

To compute (or approximate) the variance of $\hat{z}^{(k)}(t)$ is not trivial given that the receiver does not know *a priori* which set of users are active. The Markovian assumption, however, allows us to compute the so-called predicted variance by using the state estimate from the previous slot. Specifically, we propose:

$$\sigma_k^2 = \sigma^2 + \sum_{\substack{\gamma_{n-1}^{(j)} = 1 \\ j \neq k}} (1 - q^{(j)})\rho_{kj}^2 A^{(j)2} + \sum_{\substack{\gamma_{n-1}^{(i)} = 0 \\ i \neq k}} p^{(j)}\rho_{kj}^2 A^{(i)2}$$

where σ^2 is the channel noise variance, $q^{(j)}$ is the state transition probability from 1 to 0 for user j, while $p^{(i)}$ is the state transition probability from 0 to 1 for user i. Stacking the output for $t = 1, \ldots, T$, we have

$$\mathbf{w}^{(k)} = \mathbf{s}_n^{(k)}\gamma_n^{(k)} + \hat{\mathbf{z}}^{(k)} = A^{(k)}\mathbf{b}_n^{(k)}\gamma_n^{(k)} + \hat{\mathbf{z}}^{(k)}$$

We have a Gaussian vector whose mean value is either zero or $A^{(k)}\mathbf{b}_n^{(k)}$ depending on the value of $\gamma_n^{(k)}$. The unknown mean value does not admit a simple likelihood ratio test regarding $\gamma_n^{(k)}$. We propose the statistic $T^{(k)}(\mathbf{y}_n) = \mathbf{w}^{(k)\prime}\mathbf{w}^{(k)}$. The justification is two fold. First, it circumvent the estimation (or marginalization) of the unknown symbol vector $\mathbf{b}_n^{(k)}$. Second, this statistic is a uniformly most powerful invariant (UMPI) test statistic for testing mean value (zero vs nonzero) of a Gaussian vector [1]. Further, $T^{(k)}(\mathbf{y}_n)$ follows the scaled chi-square or noncentral chi-square distribution depending on the value of $\gamma_n^{(k)}$. A two-state HMM tracker for $\gamma_n^{(k)}$ could be constructed given the calibration of $T^{(k)}(\mathbf{y}_n)$ under both hypotheses. In the absence of traffic information, detection of active users could be done using a one-shot UMPI test which amounts to the simple thresholding $T^{(k)}(\mathbf{y}_n)$.

5 Simulation

Packets were generated for a total number of 20 users, with both on and off periods being geometrically distributed with mean 10 and 100 respectively. Near far effect was considered by placing the user power evenly in the log space with power range of $20dB$. The SNR shown in the plot was that for the user with minimum power. Packet length was 128 and the error correction code can correct up to 8 bit errors. The length of the spreading code was 31. First notice that the performance gain of utilizing the true model against the full model. Second, the one-shot matched filter based tracker does not perform much better than the full model DD. However, when the traffic predictability was used, the HMM based two stage MUD performs almost as well as the true model. This is because the inaccuracy of the Gaussian assumption for the MAI is well compensated via the utilization of the state continuity imposed by the traffic burstiness.

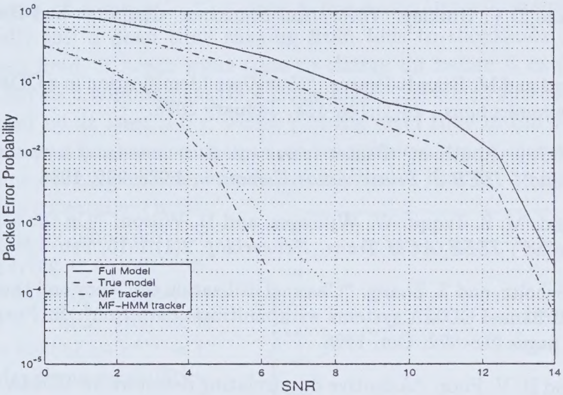


Figure 6: Performance of suboptimal HMM tracker using MF front end.

A very important issue when using the suboptimal tracker is the control of the false alarm (an inactive user being errantly claimed as active) and miss detection (excluding an active user) when inferring about the set of active users. A miss detection will result a sure packet loss. Besides, the interference of the missed user on the reception of other active users would not be mitigated at the second stage MUD hence could further hinder the reception performance. Thus the penalty imposed by the miss detection is much more severe than a false alarm, whose only effect is the increase of noise variance. This implies that the minimum error probability state estimate might not yield the best performance in terms of packet reception. A good threshold should be more stringent in controlling the miss detection. The same issue exists for the one-shot detection using only $T^{(k)}(\mathbf{y}_n)$. Finding out the optimal threshold that yields the best performance in terms of packet reception does not appear to be analytically tractable. Numerical simulations could be carried out to search for a best threshold that gives rise to the minimum packet error probability. Further, extensive simulations show that with a fixed threshold, the miss detection would eventually dominate the packet error probability as the SNR increases. This could be attributed to the fact that Gaussian approximation tends to work poorly for higher SNR. A better strategy therefore is to adapt the threshold according the to the SNR level, thus control the miss detection to a level comparable to the error reception caused by bits error uncorrectible by the error control code.

6 Conlusions

This paper studies the possibility of utilizing source level traffic predictability for the purpose of receiver design in a packet-switching network. The system considered is the random access/CDMA slotted packet switching networks and we investigate the implementation of a decorrelating detector. Conventional DD which assumes all users are active is shown to perform poorly under typical network traffic condition. A two stage MUD is proposed where the first stage of detecting the set of active users in a packet/CDMA system is investigated in detail. Simulation confirms that, for the simple MF front end, the utilization of traffic information in active user detection does provide substantial performance gain in terms of packet reception.

References

[1] D.R. Cox and D.V. Hinkley. *Theoretical Statistics.* Chapman and Hall Ltd, New York, 1974.

[2] K.W. Halford and M. Brandt-Pearce. "New-User Identification in a CDMA System". *IEEE Trans. Communications.*, 46(1):144–155, January 1998.

[3] R. Jain and S. A. Routhier. "Packet trains: measurements and a new model for computer network traffic". *IEEE J. Select. Areas Commun.*, 4:986–995, 1986.

[4] W. E. Leland, M. S. Taqqu, W. Willinger, and D. Wilson. "On the self-similar nature of ethernet traffic". *IEEE/ACM Trans. Networking*, 2(1):1–15, Feb. 1994.

[5] Q. Liu, A. Scholtz, and Z. Zhang. "Channel utilization efficiency and throughput of random access multichannel CDMA systems with multiuser detection". In *Proc. of the 15th IEEE MILCOM*, pages 918–922, Oct. 1996.

[6] U. Mitra and H. V. Poor. "Adaptive decorrelating detectors for CDMA systems". *Wireless Person Communications Journal*, 2:256–290, Dec. 1995.

[7] V. Paxson and S. Floyd. "Wide area traffic: the failure of Poisson modeling". *IEEE/ACM Trans. Networking*, COM-35(9):877–887, September 1987.

[8] L. Rabiner. "A tutorial on hidden Markov Models and selected applications in speech recognition". *IEEE Proceedings*, 77(2):257–285, Feb. 1989.

[9] P.B. Rapajic. "Performance analysis of slotted ALOHA/CDMA system with adaptive MMSE receivers". *IEICE Trans. Fundamentals*, E80-A(12):2485–2492, Dec. 1997.

[10] D. Raychaudhuri. "Performance analysis of Random Access Packet-Switched Code Division Multiple Access systems". *IEEE Trans. Communications*, 29(6):895–901, 1981.

[11] A. M. Viterbi. "Approximate analysis of time-synchronous packet networks". *IEEE J. Select. Areas Commun.*, 4:879–890, Sep. 1986.

[12] W. Willinger, M. S. Taqqu, R. Sherman, and D. V. Wilson. "Self-similariy through high-variability: statistical analysis of ethernet LAN traffic at the source level". *IEEE/ACM Trans. Networking*, 5(1):71–86, Feb. 1997.

[13] W. Wu and K. Chen. "Identification of active users in synchronous CDMA multiuser detection". *IEEE J. Select. Areas Commun.*, 16(9):1723–1735, Dec. 1998.

Optical Channels With Dead Time*

Aamod Khandekar and Robert McEliece
Department of Electrical Engineering
California Institute of Technology
Pasadena, CA 91125
{aamod,rjm}@systems.caltech.edu

1 Introduction

One of the problems of communicating over a free-space optical channel is that the transmitter typically has a dead time ranging from tens to thousands of bit periods, i.e. once it has been fired, it needs some time to charge up before it can be refired [1]. Thus, we have to devise a communication scheme under the constraint that any two 1's need to be separated by at least d 0's, where d is the dead time in bit periods. In this paper, we find the capacity of such a channel under noiseless conditions and analyze the performance of some existing and new modulation schemes that try to achieve it.

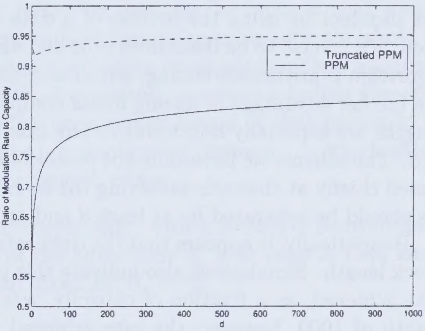


Figure 1: Fraction of Capacity achieved by the Modulation Schemes PPM and Truncated PPM

The capacity of a constrained noiseless channel is given by Theorem 1 in Shannon [3]. Applying it to our constraint, we get the capacity in nats/bit period to be $C(d) = -\ln x$, where x is the unique positive root of $x + x^{d+1} = 1$.

We now consider some modulation schemes and see to what extent they achieve the capacity. The first of these is Pulse Position Modulation(PPM). In M-ary PPM, the data is partitioned into blocks of size $\log_2 M$. If such a block represents the integer k in binary, it is mapped to a block of size $M + d$ with a 1 in the k'th position and 0's everywhere else. It is our understanding that PPM, in particular 256-PPM, is a scheme currently being considered for NASA planetary telecommunication systems.

The next scheme that we consider is a modified version of PPM which we call truncated PPM. Here, we map a block of $\log_2 M$ bits, denoting the integer k in binary, to a block of size $k + d$ with a 1 in the k'th position and 0's everywhere else.

Finally, we shall describe a scheme based on arithmetic coding following the approach of [4] that achieves capacity to any specified degree of accuracy. We also describe simulation results to support this claim.

2 Analysis of PPM and Truncated PPM

In PPM, a block of data at the input contains $\ln M$ nats of information and takes $M + d$ time slots to transmit. Hence the rate R in nats/bit period achieved by this scheme is $R = \frac{\ln M}{M+d}$. We maximize this function over M to get the optimum rate R_{PPM} over different block sizes. We find R_{PPM} to be the solution to $e^{-(R_{\text{PPM}}d+1)} = R_{\text{PPM}}$. The solution of this equation can be shown to tend to the capacity in the limit of large d.

---

*This work was partially supported by NSF grant no. CCR-9804793, and also by JPL, Caltech, under a contract with NASA.

In truncated PPM, a block of data at the input still contains $\ln M$ nats of information but on an average takes only $M/2+d$ time slots to transmit. Hence the rate R achieved in this case is $R = \frac{\ln M}{M/2+d}$. Here, maximizing over M yields the optimum rate R_{TPPM} to be the root of $e^{-(R_{\text{TPPM}}d+c)} = R_{\text{TPPM}}$, where $c = 1 - \ln 2 = 0.307$. Obviously, R_{TPPM} is strictly greater than R_{PPM}, and hence also tends to the capacity in the limit of large d. Figure 1 shows the fraction of capacity achieved by PPM and its truncated version as a function of d. We can see from this figure that truncated PPM is a significant improvement over PPM.

3 An Arithmetic Coding based Modulation Scheme

In addition to telling us the capacity of our channel, Shannon also tells us how this capacity is to be achieved. We can think of our language as a free language (see [2])on two symbols, a symbol a consisting of a single 0, and a symbol b consisting of a 1 followed by d 0's. Theorem 8 in [3] then tells us that to achieve capacity, we should transmit symbols iid, with a having probability x, x being the root of $x + x^{d+1} = 1$. Our problem is thus to map a maximum entropy binary input stream (any input stream can be compressed to such a stream) to a stream of a's and b's with the requisite statistics.

The inverse of this problem is a data compression problem, and hence we can solve our problem by using the inverse of a data compression algorithm on this stream. We need this inverse to be itself invertible and also well defined given only x. An appropriate algorithm is arithmetic coding, which is optimal in the limit of infinite block size and has the further advantage of having linear complexity with respect to block size. Large block lengths are especially important in our case because x is very close to 1 for large values of d. The scheme we have thus got is almost identical to the one used in [4], where it was aimed chiefly at channels satisfying the so-called d, k constraint, i.e. that two consecutive 1's should be separated by at least d and at most k 1's.

Heuristically, it appears that the data rate should tend to the capacity with increasing block length. Simulations also indicate this to be the case. For a block length of 100, the rate achieved, as a fraction of capacity, was above 0.94 for all values of d. For a block length of 1000, however, the rate achieved was above 0.993 times the capacity for all values of d, and for a block length of 10000, the rate was above 0.9993 times the capacity for all values of d. These numbers seem to indicate that the difference between the rate achieved and the capacity, as a fraction of capacity, decreases almost proportionally as the block length increases.

References

[1] Jon Hamkins,"The Capacity of APD-detected PPM," *Preprint.*

[2] Aamod Khandekar and Robert McEliece, "The Discrete Noiseless Channel Revisited," *To appear in the Proceeding of the ISCTA '99*

[3] C. E. Shannon, *The Mathematical Theory of Communication.* Urbana, Illinois: University of Illinois Press, 1963, 1998.

[4] Nigel Martin, Glen Langdon and Stephen Todd,"Arithmetic Codes for Constrained Channels," *IBM J. Res. Develop.* vol. 27, no. 2, pp. 107-115, 1983.

AWGN Coding Theorems for Serial Turbo Codes[1]

Hui Jin and Robert J. McEliece
California Institute of Technology
Pasadena, California USA
E-mail: `(hui, rjm)@systems.caltech.edu`

Abstract. *In this paper we discuss AWGN coding theorems for ensembles of serial turbo codes. We first prove a general theorem, which shows that under very general conditions, the ensemble word error probability approaches zero as the block length approaches infinity, provided the bit signal-to-noise ratio exceeds some threshold. After that, we prove this threshold is finite for most serial turbo codes, by applying an upper bound ([4]) on weight enumerators of convolutional codes. The conditions under which coding theorem for those ensembles of codes holds will be clarified and the interleaver coding gain will be identified.*

1 Introduction

Serial turbo codes have been considered as a valid, in some cases superior, alternative to classical Berrou et al. turbo codes. Our specific goal here is to prove a coding theorem for the ensemble of a serial turbo code, where the ensemble is obtained by using the uniform interleaver technique [1]. Throughout this paper, we restrict ourselves to AWGN channel with ML decoding. The tools we applied are union bound and weight enumerator. Previously in [3] we have analyzed RA codes, by deriving the input-output weight enumerator (IOWE), from which we could compute the signal-to-noise ratio threshold by a general technique in [2]. Difficulty arises when component codes get complex, making it intractable to even closely estimate the IOWE. Fortunately, to prove a coding theorem, the exact IOWE isn't indispensable, instead, a good upper bound of that proves to be sufficient. The main contribution of this paper is the following theorem 1.1, which shows that ensembles of most serial turbo codes are "good" codes. [2]

Theorem 1.1. *For the ensemble of a serial turbo code, with a recursive convolutional code as the inner code, if the free distance of the outer code* d_o^* *is at least 3, then there exists a positive number* γ_o*, such that for any fixed* $E_b/N_o > \gamma_o$*,*

$$P_W = O(n^{-\lfloor \frac{d_o^*-1}{2} \rfloor + \epsilon}) \,. \tag{1.1}$$

for arbitrary $\epsilon > 0$.

2 An AWGN Coding Theorem Based on the Weight Enumerator

Consider a linear (n, k) block code C with code rate $R_c = k/n$ and minimum distance h_m. The union bound on the word error probability P_W for C over a memoryless binary-input AWGN channel with ML decoding can be written as:

$$P_W \leq \sum_{h=h_m}^{n} P_W(h) \tag{2.1}$$

---

[1] This work was supported by NSF grant no. CCR-9804793, and grants from Sony and Qualcomm.

[2] "good" in the sense as MacKay defined in [5].

where $P_W(h) = A_h e^{-R_c E_b/N_o}$ is the probability that a codeword is decoded to some other codeword with hamming distance h, and A_h is the number of codewords with output weight h.

To this end, we use notations: *spectral shape* $r_n(\delta) \overset{\text{def}}{=} (\ln A_h)/n$; $D_n \overset{\text{def}}{=} \ln^2 n$; *code threshold* $c_o \overset{\text{def}}{=} \sup_{0<\delta\leq 1} r_n(\delta)/\delta$; and $Z^{(n)}(D) \overset{\text{def}}{=} \sum_{h=h_m}^{D} A_h$.

Theorem 2.1. *Assume* $Z^{(n)}(D_n) = O(n^{\beta})$, *if* c_o *is finite, then for* $E_b/N_o > c_o/R_c$, $P_W = O(n^{\beta})$. *Further if* $\beta < 0$, *then* P_W *approaches zero as* n *approaches infinity.*

In the theorem, $Z^{(n)}(D_n)$ is determined by the minimum distance of the ensemble. And the condition whether c_o is finite should be checked by the following property.

Property 2.1. *Code threshold* c_o *is finite if and only if the spectral shape* $r_n(\delta)$ *satisfies* $\limsup_{\delta\to 0} r_n(\delta)/\delta < \infty$, *as* n *becomes large.*

3 Proof Outline of Main Theorem 1.1

We use two upper bounds on IOWE, which are essentially due to Kahale and Urbanke [4], to prove theorem 1.1. First, we prove code threshold c_o is finite for $d_o^* \geq 3$, by applying property 2.1. Second, we show that $Z^{(n)}(D_n) = O(n^{-\lfloor \frac{d_o^*-1}{2} \rfloor + \epsilon})$. Now, theorem 1.1 follows from theorem 2.1.

4 Remarks

By no means the thresholds derived by classical union bound are the *best* possible. In the following table, we compare those thresholds derived by union bound with those by Gallager-Divsalar (GD) bound for RA codes.

q	R	UB:γ_q (dB)	GD:γ_q (dB)	Shannon (dB)
3	1/3	2.200	0.792	−0.495
4	1/4	1.928	−0.052	−0.794
5	1/5	1.798	−0.480	−0.963
6	1/6	1.721	−0.734	−1.071

References

1. S. Benedetto, D. Divsalar, G. Montorsi, and F. Pollara. Serial concatenation of interleaved codes: Performance analysis, design and iterative decoding. *IEEE Trans. on Info. Theory*, 44:909–926, May 1998.
2. D. Divsalar, S. Dolinar, H. Jin, and R. J. McEliece. AWGN coding theorems from ensemble weight enumerators. In Preparation, 1999.
3. D. Divsalar, H. Jin, and R. J. McEliece. Coding theorems for "turbo–like" codes. In *Proc. 36th Allerton Conf. on Comm., Control and Computing*, pages 201–210, Allerton, IL, September 1998.
4. N. Kahale and R. Urbanke. On the minimum distance of parallel and serial concatenated codes. In *ISIT 1998*, page 31, MIT, Cambridge, 1998.
5. D. J. C. MacKay. Good error correcting codes based on very sparse matrices. *IEEE Trans. on Info. Theory*, 45:399–431, March 1999.

Linear Codes over $\mathbb{Z}/(2^k)$ of Constant Euclidean Weight

Jay A. Wood
Department of Mathematics, Computer Science & Statistics
Purdue University Calumet
Hammond, Indiana 46323–2094
`wood@calumet.purdue.edu`

Over finite fields, any linear code with constant Hamming weight is a replication of simplex (i.e., dual Hamming) codes [1]. Recently, Carlet [2] proved a similar result for linear codes of constant Lee weight over $\mathbb{Z}/(2^k)$. In this short paper we prove the existence of linear codes of constant Euclidean weight over $\mathbb{Z}/(2^k)$.

Throughout this paper, the ground ring will be $R = \mathbb{Z}/(2^k)$. A *linear code* of length n is a submodule of R^n, and it will be viewed as an underlying finite R-module M equipped with a linear embedding $\lambda : M \to R^n$ given by n *coordinate functionals* $\lambda_1, \dots, \lambda_n \in M^\sharp := \mathrm{Hom}_R(M, R)$. Taking representatives satisfying $-2^{k-1} < t \leq 2^{k-1}$, the *Euclidean weight* of $t \in R$ is $w(t) = t^2 \in \mathbb{Z}$. The Euclidean weight of $x \in M$ is $w(x) = \sum w(\lambda_i(x)) \in \mathbb{Z}$. A linear code has *constant Euclidean weight* $L > 0$ if $w(x) = L$ for all nonzero $x \in M$.

Any finite R-module M determines nonnegative integers $l_1, l_2, \dots, l_k$ so that

$$M \cong \bigoplus_{i=1}^{k} \left(\mathbb{Z}/(2^i)\right)^{l_i}, \tag{1}$$

as R-modules. Set $e_0 = 1$ and $e_i = l_1 + 2l_2 + \cdots + (i-1)l_{i-1} + i(l_i + \cdots + l_k) - i$, for $1 \leq i \leq k$. The R-module $M^\sharp$ admits a filtration (its upper Loewy chain)

$$M^\sharp \supset 2M^\sharp \supset 4M^\sharp \supset \cdots \supset 2^{k-1}M^\sharp \supset 2^k M^\sharp = 0. \tag{2}$$

For $\lambda \in M^\sharp$, let $\nu(\lambda)$ be the largest exponent such that $\lambda \in 2^{\nu(\lambda)} M^\sharp$.

THEOREM. *Let M be any module over $R = \mathbb{Z}/(2^k)$ of the form (1), with $l_k \geq 1$. As coordinate functionals, use every nonzero linear functional $\lambda \in M^\sharp$, with multiplicity*

$$m_{\nu(\lambda)} = \sum_{i=0}^{\nu(\lambda)} 2^{e_i}.$$

Then this linear code has cardinality, length, and constant Euclidean weight given by

$$|M| = 2^{l_1 + 2l_2 + \cdots + kl_k}, \; n = \frac{|M|}{2^{k-1}}\left(3 \cdot 2^{k-1} - 1\right) - m_{k-1}, \; \text{and } L = 2^{2k-2}|M|.$$

PROOF. First note that since we assume $l_k \geq 1$, 2^k divides $|M|$, and n is an integer.

Partially supported by Purdue University Calumet Scholarly Research Awards.

Take any nonzero $x \in M$. Using the definition of $m_{\nu(\lambda)}$ and the filtration (2),

$$w(x) = \sum_{\lambda \in M^\sharp} m_{\nu(\lambda)} w(\lambda(x)) = \sum_{j=0}^{k-1} 2^{e_j} \sum_{\lambda \in 2^j M^\sharp} w(\lambda(x)). \tag{3}$$

Now consider the linear functional $\check{x} : 2^j M^\sharp \to R$, $\lambda \mapsto \lambda(x)$. Suppose x has order 2^i. If $i \leq j$, the functional $\check{x}$ is zero. When $i > j$, the image $\mathrm{im}(\check{x}) = \check{x}(2^j M^\sharp) = (2^{k-i+j})$, with $|\mathrm{im}(\check{x})| = 2^{i-j}$. In the latter case, every $r \in \mathrm{im}(\check{x})$ is hit equally often by $\check{x}$, namely

$$|\ker(\check{x})| = \frac{|2^j M^\sharp|}{|\mathrm{im}(\check{x})|} = 2^{l_{j+1}+2l_{j+2}+\cdots+(k-j)l_k-i+j}$$

times. It is straight forward to show that, for Euclidean weight,

$$3 \sum_{r \in (2^\mu)} w(r) = 2^{k+\mu-1}\left(2^{2k-2\mu-1}+1\right). \tag{4}$$

Remembering that x has order 2^i and using $\mu = k - i + j$ in (4), we conclude that

$$3 \sum_{\lambda \in 2^j M^\sharp} w(\lambda(x)) = \begin{cases} 0, & i \leq j, \\ 2^{l_{j+1}+2l_{j+2}+\cdots+(k-j)l_k}\left(1+2^{-2i+2j+1}\right)2^{2k-2}, & i > j. \end{cases}$$

This last expression allows us to rewrite (3) as

$$3w(x) = \sum_{j=0}^{i-1} 2^{e_j+l_{j+1}+2l_{j+2}+\cdots+(k-j)l_k}\left(1+2^{-2i+2j+1}\right)2^{2k-2}. \tag{5}$$

Observe that the complicated exponent in (5) simplifies to

$$e_j + l_{j+1} + 2l_{j+2} + \cdots + (k-j)l_k = \begin{cases} 1 + \sum_{q=1}^k q l_q, & j = 0, \\ -j + \sum_{q=1}^k q l_q, & j > 0. \end{cases}$$

Since $\log_2 |M| = \sum_{q=1}^k q l_q$, (5) simplifies to

$$3w(x) = 2|M|\left(1+2^{-2i+1}\right)2^{2k-2} + \sum_{j=1}^{i-1} |M|\, 2^{-j}\left(1+2^{-2i+2j+1}\right)2^{2k-2}.$$

By summing the geometric series and simplifying, we obtain the constant weight

$$w(x) = 2^{2k-2}|M|.$$

The remaining claims of the theorem can be verified easily. □

REMARK. It can also be proved that the above is essentially the only way to obtain linear codes of constant Euclidean weight over $\mathbb{Z}/(2^k)$. Up to code equivalence (permutations and sign changes of the λ_i) and padding with zeros, every linear code on M of constant Euclidean weight is a replication of the one in the Theorem. (If the multiplicities m_i are not relatively prime, first scale them by their GCD. This gives a "minimal model.") This result will appear in subsequent work of the author.

References

[1] A. Bonisoli, *Every equidistant linear code is a sequence of dual Hamming codes*, Ars Combin. **18** (1984), 181–186.

[2] C. Carlet, *One-weight $\mathbb{Z}_4$-linear codes*, preprint, 1998.

URL: `http://www.calumet.purdue.edu/public/math/wood/`

Space-Time Turbo Codes

Youjian Liu and Michael P. Fitz
Department of Electrical Engineering, The Ohio State University
`liuy@eemail.eng.ohio-state.edu; fitz.7@osu.edu`

Introduction: Recently, there has been a surge of interests in the design of the so called "space–time" codes which take advantage of both the spatial diversity provided by multiple antennas and the temporal diversity available with time–varying fading. The previous proposed space–time codes were mainly based on trellis codes or orthogonal block codes. In those codes, the number of transmitter antennas is fixed for each code construction. We propose a new class of scalable space–time codes based on turbo codes or turbo trellis codes [1]. They will be referred as space–time turbo codes (STT) in the sequel. The scalability implies that the code rate and number of transmitter antennas can be easily adapted to different design requirements without redesigning the main part of the code. We present the encoder, channel model followed by the design criteria, simulation results and conclusions.

Encoder and Channel Model: The encoder of the proposed space–time turbo code consists of a turbo code encoder or a turbo trellis code encoder followed by a symbol interleaver and multiplexer (Fig. 1). Puncturing the code allows a variety of code rates and the number of transmitter antennas can be arbitrary as long as it is not too close to the free distance of the turbo code.

We show the performance of three example codes in the simulation results. The turbo code part of Example 1 and 2 are rate 1/3 BPSK turbo codes with constraint length 3 and 4 component convolutional code respectively. The turbo code part of Example 3 is a rate 2/2 QPSK turbo trellis code (Fig. 2) modified from a rate 1/2 BPSK turbo code in [1]. No puncture is used in these Examples. All the information and symbol interleavers are different samples of a uniform random interleaver. The output symbols are cyclically multiplexed to 2 transmitter antennas.

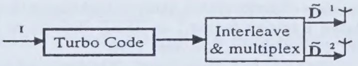


Fig. 1. Structure of Space–Time Turbo Encoders with 3 Component Encoder and 2 Transmitter Antennas.

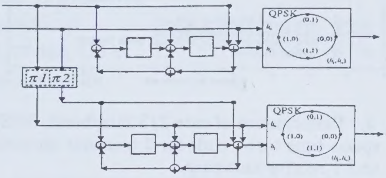


Fig. 2. Turbo Trellis Encoder with 4 states component code, Code Rate=1

The channel model is a multiple transmitter antennas and multiple receiver antennas flat slow Rayleigh fading channel. The fading paths between different antennas are assumed to be independent. Assuming perfect knowledge about fading coefficients, Pearl's belief propagation algorithm [2] is used as a sub–optimal iterative decoding method.

Design Criteria: Design criteria [3], [4], [5] have been proposed for space-time codes in a variety of environments. In Rayleigh fading, each possible code word difference in a coded modulation produces a 'signal' matrix which is a function of both the code word difference and the channel correlation. The diversity provided by a code word difference is given by the rank of this 'signal' matrix and the effective product measure [5]. The proposed STT use random symbol interleavers, so the code word differences are spread among all the transmitter antennas, which results in full rank and large effective product measure 'signal' matrix with high probability.

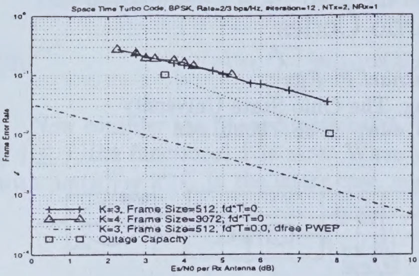


Fig. 3. Performance of rate 2/3 bits/baud, BPSK space–time turbo code for 2 transmit antennas and 1 receive antennas.

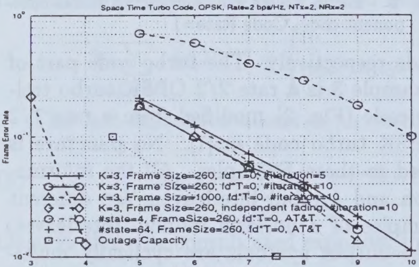


Fig. 4. Performance of rate 2 bits/baud, BPSK space–time turbo code for 2 transmit antennas and 2 receive antennas.

Simulation Results: The simulated performances of Example 1 and 2 STT codes are shown in Fig. 3 (K is constraint length) for quasi–static fading channel. The performances are close to each other. This means that we can choose very simple component codes without losing much in performance. At a frame error rate of 10%, the codes are 1.5 dB away from outage capacity [6]. In addition, Fig. 3 has a plot of the sum of the 5 pair-wise error probabilities (PWEP) for the five lowest binary weight (16) codewords of Example 1. This curve has the same slope as the Example STT code which implies that the STT is obtaining the full diversity. The constant term is off by a multiple of approximately 20 and further investigation of high weight code words is needed to acertain exactly why.

Fig. 4 shows the simulation results of Example 3 compared with two AT&T space–time codes [4]. The performances of the STT code is close or better than AT&T's 64 states code. At a frame error rate of 10%, the performance of STT codes is about 1.5 dB away from the outage capacity. From the performance associated with different number of iterations, we observe that the number of iterations can be as small as 5 without losing much in performance. Simulation for time independent fading shows that the gain over the quasi-static fading case is about 6dB at a frame error rate 0.01, which implies that Example 3 has a significantly rich structure to take advantage of both the temporal and spatial diversity.

What is common in Fig. 4 and 3 is that the increase of information frame size either improves or does not degrade the frame error rate. For trellis space–time code, the frame error rate will degrade significantly for the case of frame size 1000 because there is no interleaver gain.

Conclusions: We proposed a new class of space–time turbo codes. With high probability, this class of codes will take full advantage of both space and temporal diversity. Simulations of a few ad-hoc selected examples demonstrate good performance especially for medium to long frame sizes. Further improvements are likely with a principled approach to code design.

REFERENCES

[1] D. J. Divsalar and F. Pollara, "Turbo trellis coded modulation with iterative decoding for mobile satellite communications," in *IMSC 97*, June 1997.

[2] R. J. McEliece, D. J. MacKay, and Jung-Fu Cheng, "Turbo decoding as an instance of Pearl's "belief propagation" algorithm," *IEEE J. Select. Areas Commun.*, vol. Vol. 16, pp. 140–152, Feb. 1998.

[3] J. C. Guey, M. P. Fitz, M. R. Bell, and W. Y. Kuo, "Signal design for transmitter diversity wireless communication systems over Rayleigh fading channels," in *Proc. of IEEE Vehicular Technology Conference*. Atlanta, GA, 1997, vol. 1, pp. 136–140.

[4] V. Tarokh, N. Seshadri, and A. R. Calderbank, "Space-time codes for high data rate wireless communication: performance criterion and code construction," *IEEE Trans. on Info. Th.*, vol. 44, no. 2, pp. 744–765, Mar. 1998.

[5] M. P. Fitz, J. Grimm, and S. Siwamogsatham, "A new view of performance analysis techniques in correlated Rayleigh fading," in *IEEE WCNC, to appear*, New Orleans, LA, Sep. 1999.

[6] G. J. Foschini and M. J. Gans, "On limits of wireless communications in a fading environment when using multiple antennas," *Wireless Personal Communications*, vol. 6, pp. 311–335, March 1998.

Turbo Decoding of Concatenated Space-Time Codes

Krishna R. Narayanan
Department of Electrical Engineering
Texas A&M University,
College Station, TX 77843, USA
Email: krishna@ee.tamu.edu

Abstract

We propose a class of codes called turbo space-time codes, which combine the principles of turbo coding and space-time coding. It is first shown that space-time codes can be realized as recursive convolutional codes. This fact is then exploited to design parallel and serial concatenated codes using space-time codes as component codes. Finally, iterative decoding of such concatenated codes is shown to significantly improve the power efficiency of space-time codes.

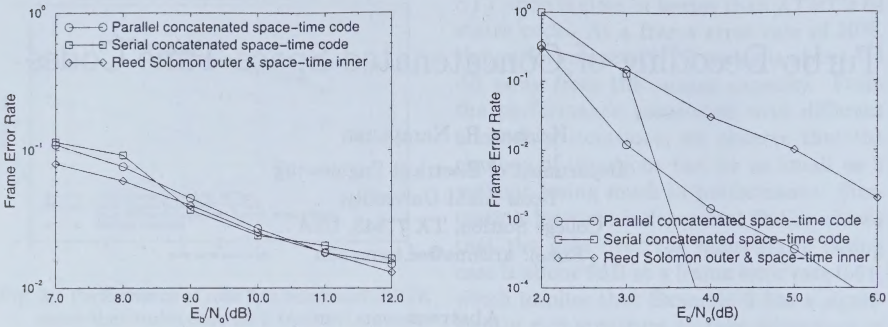
I Introduction

Space-time codes are designed for use with multiple transmit antennas and exploit the available spatial diversity to improve the performance. Concatenated space-time codes which guarantee a diversity of r_1 under quasi-static fading and a diversity of $r_2 > r_1$ under fast fading have been designed in [3]. Tarokh *et al*'s work was mainly targeted at delay-sensitive applications and, hence, only short block lengths were considered.

In this paper, we consider the design of parallel and serial concatenated smart-greedy codes based on the turbo coding principle and iterative decoding of such codes. The codes are constructed using space-time codes as component codes and are shown to provide excellent performance for fast fading channels.

II Recursive Space-Time Codes

Consider the space-time encoder. The data sequence is first demultiplexed in to p parallel streams and the set of p bits at time k is denoted by $\boldsymbol{x}_k = [x_{1k}\ x_{2k}\ \ldots\ x_{pk}]$. At each time instant a group of p bits selects one of the 2^p possible branch transitions. The outputs along the transition are transmitted using n antennas. The encoder is equivalent to n separate rate-1 encoders followed by a memoryless mapper. The output of the ith binary encoder at time k is a $p-$tuple denoted by $\boldsymbol{y}_{i,k} = [y_{1,i,k}\ y_{2,i,k}\ \ldots y_{p,i,k}]$. The D transform of $\boldsymbol{y}_i$ for $1 \leq i \leq n$ can be expressed in matrix form as $\boldsymbol{Y}_i(D) = \boldsymbol{X}(D)\ \boldsymbol{G}(D)$ where the ith row of $\boldsymbol{G}(D)$ is $\boldsymbol{G}_j(D) = [g_{j,1,i}, g_{j,2,i}, \ldots, g_{j,p,i}]$ where $g_{j,k,i}(D)$ is the generator polynomial linking the jth input stream to the kth output stream for the ith transmit antenna. Now, consider a generator matrix $\boldsymbol{G}'(D)$ obtained by dividing the jth row of $\boldsymbol{G}(D)$ by any polynomial $T_j(D)$ such that $deg(T_j(D)) < \nu_j$. Then, the new generator matrix represents a recursive encoder for generating the same set of code words as that of the original space-time code.



(a) (b)

Figure 1: Performance of turbo space-time codes on (a) Quasi-static Rayleigh fading (b) Uncorrelated Rayleigh fading

Since recursive encoders can provide significant interleaving gain when used in parallel and serial concatenated schemes, recursive space-time codes can provide significant interleaving gains too. In parallel concatenated schemes, space-time codes can be used as component encoders while in serial concatenated schemes, the recursive space-time codes can be used as the inner code. It is important to note that in this serial concatenation set up, the function of the space-time inner code is mainly to guarantee a minimum diversity in a quasi-static fading case. The majority of the diversity advantage in rapidly fading channels is a result of the interleaving gain from the concatenated code.

III Results

The performance of parallel and serial concatenated smart-greedy codes for 2 transmit antennas and one receive antenna for a rate of 0.5 b/s/Hz using BPSK is shown in Figure 1. The parallel concatenated code is a parallel concatenation of two 4-state space-time codes. A data block length and hence the interleaver length of 1000 bits and an $S-$random interleaver with $S = 15$ was used in the simulations. The serial concatenated code considered is the concatenation of a rate-1/2, 4-state outer convolutional code with generator polynomials $[1+D+D^2, 1+D^2]$ and a 2-state inner space time code using BPSK. The performance of a 0.37 b/s/Hz smart-greedy code from [3] which is the concatenation of an outer RS code and an inner space-time code is also shown. We can see from the figures that the concatenated codes designed in this paper significantly outperform the smart-greedy codes in [3] for the fast fading cases. For the quasi-static fading case, the frame error rate performance of these codes is almost the same.

References

[1] V. Tarokh, N. Seshadri, and A. R. Calderbank, "Space-time codes for high data rate wireless communication: Performance criterion and code construction," *IEEE Trans. Inform. Theory*, pp. 744–765, March 1998.

Soft Output and Iterative Stack Decoding *

Ravi Sivasankaran, Steven W. McLaughlin †
School of Electrical and Computer Engineering.
Georgia Institute of Technology, Atlanta, GA 30332.
ravi, swm@ece.gatech.edu

Abstract

In this paper, we present a method for soft-in/soft-out sequential decoding of recursive systematic convolutional codes. The proposed decoder uses two stacks, and lends itself to the generation of soft outputs. The decoder is easily incorporated into the iterative "turbo" configuration. At moderate to high SNRs (above 2.0 dB), the performance of the proposed decoder is within 1.5 dB of the BCJR algorithm for a 16-state, R = 1/3, recursive code. The complexity of the proposed decoder, on the other hand, asymptotically decreases as 1/(number of states), thus, providing a good trade-off between computational burden and performance.

1 Introduction

In this paper we will consider the 'stack' based sequential decoding techniques. We propose an algorithm, the Twin Stack decoder, which is based on the stack decoding technique, for decoding recursive systematic convolutional (RSC) codes. We will show how the proposed algorithm is capable of near ML/MAP performance, and how this performance can be acheived at a fraction of the complexity of the ML technique. We will also show how the proposed decoder lends itself to the generation of soft outputs on the received information sequence in the form of Log Likelihood Ratios (LLR). Finally, we will show how the soft output Twin Stack decoder can be used in an iterative configuration to decode the parallel concatenated RSC codes ('turbo' codes). It was observed that the Twin Stack decoder acheives performance as close as 1.0 dB to the BCJR decoder [1] at Bit Error Rates of $\leq 10^{-5}$. For complete details on the Twin Stack decoder see [2].

2 Results

The Twin Stack decoder has the capability of retaining both transition information into any state at any instant of time, which makes it particularly suited for decoding RC codes. The decoder is capable of acting as a soft-out decoder, and the post processing unit that generates the LLR on the input sequence approaches the MAP decoding technique if the probability of the unexpanded paths in the stack are negligible, and/or at high SNRs.

The performance of the soft-in/soft-out Twin Stack decoder as a component decoder in the iterative decoding of a 16-State, $R = 1/3$, turbo code is shown in Figure 1, and is compared with the classical turbo decoder. We find that the performance of the BCJR-based decoder is better by about 1.5 dB at BERs $\geq 10^{-4}$, but the performance

---

*This work was supported in part by NSF Grant NCR-9702024.
†Corresponding author

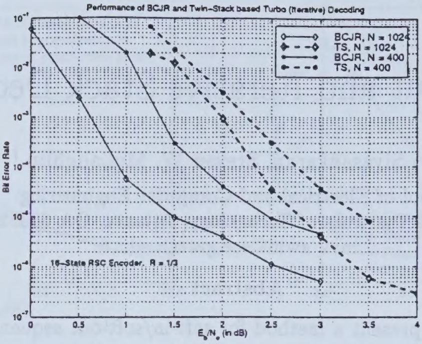


Figure 1: Performance of BCJR-based and Twin-Stack based Turbo decoding.

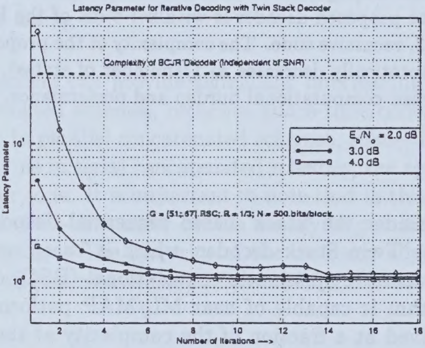


Figure 2: Variation of Latency Parameter with Iterations.

difference narrows progressively for higher SNRs. In Figure 2 we show how the latency parameter changes with iterations and SNR for a Twin Stack decoder. Thus, the Twin Stack decoder gives us the necessary flexibility to trade computational burden off for performance. The performance gap between the Twin Stack and the BCJR decoder can be narrowed considerably by using higher constraint length encoders, without the penalty of a steep increase in the computational burden at the decoder. The soft-out Twin Stack decoder can be made to act as the **soft-out M-algorithm** if the number of states expanded at every decoding instant is kept at $1 \leq M \leq 2^\nu$, or as the **BCJR algorithm** (MAP decoder) itself, if all the states are expanded at any decoding instant.

References

[1] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon limit error-correcting coding and decoding: Turbo-codes(1)," in *Proceedings ICC '93*, May 1993.

[2] R. Sivasankaran, and S. W. McLaughlin, "Twin Stack decoding of recursive systematic convolutional codes," *Submitted to IEEE Trans. on Comm.*, Aug. 1999.

Algebraic Geometric codes and improvements on the Gilbert-Varshamov bound *

Hiren Maharaj
Dept. of Mathematics
Pennsylvania State University
University Park, PA 16802-6401
maharaj@math.psu.edu

We announce results which are improvements of results of Niederreiter and Xing from the asymptotic theory of algebraic curves over finite fields with many rational points. We explain in some detail how these results imply an improvement of the Gilbert-Varshamov bound (GV bound) for linear codes over finite fields of a sufficiently large composite nonsquare order.

Let $\mathbb{F}_q$ be the finite field with q elements. By an $[n, k, d]$ linear code C over $\mathbb{F}_q$ we mean a k-dimensional $\mathbb{F}_q$ vector subspace of $\mathbb{F}_q^n$ with minimum distance d. In order to compare the performances of codes of different lengths we define the *rate* $R(C)$ of C by $R(C) = k/n$ and the *relative minimum distance* $\delta(C)$ of C by $\delta(C) = d/n$. Form the set

$$V_q = \{(\delta(C), R(C)) \in [0,1]^2 | C \text{ is a linear code over } \mathbb{F}_q\}$$

and let U_q be the closure of V_q. Manin showed that U_q has the following description. There exists a continuous function $\alpha_q : [0, 1] \rightarrow [0, 1]$ such that

$$U_q = \{(\delta, R) | 0 \leq \delta \leq 1 \text{ and } 0 \leq R \leq \alpha_q(\delta)\}. \tag{1}$$

The function α_q is decreasing in the interval $[1, 1 - 1/q]$, it is identically zero on the interval $[1 - 1/q, 1]$ and $\alpha_q(0) = 1$. However, not a single value of α_q on the interval $(0, 1 - 1/q)$ is known. Now, by definition, $(\delta, R) \in U_q$ if and only if there exists an infinite sequence of codes C_i such that $\lim_{i\rightarrow\infty}(\delta(C_i), R(C_i)) = (\delta, R)$. Since there are finitely many codes of a given length, one has $n(C_i) \longrightarrow \infty$. The set U_q contains information about the distribution of parameters of long codes. Finding good lower bounds for $\alpha_q(\delta)$ implies the existence of good long codes. There are various lower bounds for $\alpha_q(\delta)$, the best, for some time being the GV bound. The GV bound says that

$$\alpha_q(\delta) \geq 1 - H_q(\delta) \quad \text{for all } \delta \in (0, 1 - 1/q). \tag{2}$$

where $H_q(\delta) = \delta \log_q(q - 1) - \delta \log_q \delta - (1 - \delta) \log_q(1 - \delta)$.

Another lower bound, which depends on algebraic-geometric codes, is given by

$$\alpha_q(\delta) \geq 1 - \frac{1}{A(q)} - \delta \quad \text{for all } \delta \in [0, 1] \tag{3}$$

where the number $A(q)$ is the upper limit of the ratio of the maximum number of points of a algebraic curve (projective, smooth, absolutely irreducible) defined over $\mathbb{F}_q$ to the genus. Vlădut and Drinfel'd established that $A(q) \leq q^{1/2} - 1$ for all q.

---

*Research supported in part by a grant from the National Science Foundation no. DMS96-22938.

In the case that q is a square, Ihara showed that $A(q) = q^{1/2} - 1$. Consequently the bound (3) becomes $\alpha_q(\delta) \geq 1 - \frac{1}{\sqrt{q}-1} - \delta$. This lower bound is better than the GV bound on a certain interval for $q \geq 49$. Garcia and Stichtenoth proved that if q is a square, then $A(q) = q^{1/2} - 1$ can be achieved by an explicitly constructed tower of global function fields. When q is not a square the exact value of $A(q)$ is not known but lower bounds are available which complement the general upper bound. According to a result of Serre based on class field towers, we have $A(q) \geq c \log q$ with an absolute constant $c > 0$. Zink gave the best known lower bound for p^3: $A(p^3) \geq \frac{2(p^2-1)}{p+2}$. Recently, Niederreiter and Xing established the following bounds. If q is odd and $m \geq 3$ is an integer, then

$$A(q^m) \geq \frac{2q}{\lceil 2(2q+1)^{1/2} \rceil + 1}. \tag{4}$$

If $q \geq 4$ is even and $m \geq 3$ is an odd integer, then

$$A(q^m) \geq \frac{q+1}{\lceil 2(2q+2)^{1/2} \rceil + 2}. \tag{5}$$

As a consequence, they improved the GV bound for sufficiently large composite nonsquare q on a certain interval.

By using the explicit construction of ray class fields of function fields via rank one Drinfeld modules we found improvements of the bounds of Niederreiter, Xing. By $\mathrm{f}(q) = \mathrm{O}(\mathrm{g}(q))$ we mean that there is a constant $M > 0$ such that $|\mathrm{f}(q)| \leq M|\mathrm{g}(q)|$ for all sufficiently large q. Let q be a power of an odd prime. If $r < s < 2r$, then for all q sufficiently large we have

$$A(q^{rs}) \geq \frac{\sqrt{2}(r-2)}{r-1}\sqrt{s}q^{s/2} + \mathrm{O}(1). \tag{6}$$

Then for any odd integer $r \geq 3$ we have

$$A(q^r) \geq \frac{4q+4}{\left\lfloor \frac{3+\lceil 2(2q+2)^{1/2} \rceil}{r-2} \right\rfloor + \lceil 2(2q+3)^{1/2} \rceil}. \tag{7}$$

Next suppose that q is even. If $r < s < 2r$, then for all q sufficiently large we have

$$A(q^{rs}) \geq \frac{\sqrt{2}}{4}\sqrt{s}q^{s/2} + \mathrm{O}(1). \tag{8}$$

For $r \geq 5$ odd and all q sufficiently large we have

$$A(q^r) \geq \frac{2q^2}{\sqrt{2q}(q-1) + 2\lceil 2\sqrt{2q} \rceil + 4}. \tag{9}$$

For $r = 3$ and all q sufficiently large we have

$$A(q^3) \geq \frac{2q^2}{\sqrt{2q}(q-4) + 8\lceil \sqrt{2q} \rceil + 16}. \tag{10}$$

For further results and a full set of references see [1].

References

[1] W-C W. LI and H. MAHARAJ, Coverings of curves with asymptotically many rational places, in preparation.

Single-Sample-Path-Based Optimization of Markov Decision Process

Zhiyuan Ren and Bruce H. Krogh
Department of Electrical and Computer Engineering
Carnegie Mellon University
5000 Forbes Avenue
Pittsburgh, PA 15213-3890
{*rzy,krogh*}*@cmu.edu*

Abstract

We present a single-sample-path-based optimization scheme for Markov decision process (MDP) with long-run average cost when the system transition and reward structures are unknown. So-called O-potentials are estimated along a single sample path of the system and actions are applied by our adaptive optimization scheme based on the latest estimates. We prove that an optimal policy is obtained asymptotically with probability one. More importantly, we prove that optimal system performance is achieved as well, which means that the asymptotic performance of the system can not be bettered even if the system transition and reward structures are known. An example is given to illustrate our scheme.

1 Introduction

This paper concerns finite Markov decision problem with long-run average cost. There are well-developed dynamic-programming-based value iteration and on-line policy iteration algorithms for solving this problem if the transition and cost structures of the system are known [4],[12]. We are interested in giving a solution to the problem without the knowledge of these structures. This calls for an adaptive strategy.

One adaptive approach is the so-called *certainty equivalence (CE) control* [3],[9]. In this approach the system transition and reward structures are assumed to be governed by an unknown parameter. At each sampling instant, the unknown parameter is estimated first, and then the control is computed using the parameter estimate as if it were the actual parameter value (certainty equivalence). Another adaptive approach is Q-learning [13]. In this case the optimal policy is obtained from so-called Q-factors that characterize the cost as a function of the state and action. In the on-line Q-learning algorithm, Q-factors are estimated and actions are chosen based on the latest estimates. In [2], a Q-learning approach is developed for the average cost Markov decision problem, but the proof of the estimation convergence is not complete. Abounadi et al. [1] give a complete convergence proof, under the assumption that non-optimal actions are used indefinitely with a probability bounded away from zero. This means that, although the optimal Q-factors are learned with probability one, the optimal system performance is never achieved.

This paper presents an adaptive scheme that is similar to the Q-learning approach, but without the strong assumption required in [1]. We introduce the concept of *optimal direction potentials (O-potentials)* for the Markov decision process (MDP) with long-run average cost, which is an extension of the concept of *performance potentials* in [4]. O-potentials introduce flexibility in designing on-line adaptive optimization schemes for the average cost problem, which plays an important role in proving our convergence results.

We give a recursive on-line estimation scheme for the O-potentials based on stochastic approximation [5],[7]. Based on this estimation scheme, an adaptive optimiztaion rule is then presented. We prove that the optimal policy is obtained with probability one along a single sample path of the system. More importantly, our adaptive rule also achieves the optimal long-run average cost of the system, which means that the asymptotic performance of the system can not be bettered by another scheme designed even with the knowledge of the system transition and reward structures.

The remainder of the paper is organized as follows. In Section 2, we first address the standard MDP problem with long-run average cost, which leads to the formulation of our problem. Some useful results for solving the standard problem are reviewed in Section 3. We introduce O-potentials in Section 4. In Section 5, an estimation scheme for the O-potentials is given, and an adaptive optimization rule is presented based on the estimation scheme. It is proved in this section that the optimal policy as well as optimal long-run average cost are achieved by our adaptive rule. Section 6 presents a small example.

2 The Problem

We first introduce some notations and state the standard MDP problem with long-run average cost. Throughout the paper, $|A|$ denotes the cardinality of a set A and $\|\cdot\|$ denotes the infinity norm of the matrix or vector argument. A discrete time Markov chain, denoted as $\mathbf{X} = \{X_n, n \geq 0\}$, has a finite state space $\mathcal{S} = \{1, 2, \cdots, |\mathcal{S}|\}$. At each time $n \geq 0$, an action is taken from a finite action space $\mathcal{A}$ and applied to the Markov chain, where $\mathcal{A}(i) \subseteq \mathcal{A}$ is the set of actions available for state $i \in \mathcal{S}$. If an action $\alpha \in \mathcal{A}(i)$ is taken at state i, then the state transition probabilities at state i are $p^{\alpha}(i,j), j = 1, 2, \cdots, |\mathcal{S}|$, and a cost $f(i, \alpha)$ is incurred.

A stationary *policy* $\mathcal{L} \in \mathcal{E}$ is a mapping $\mathcal{L} : \mathcal{S} \to \mathcal{A}$, where for any state i, $\mathcal{L}$ specifies an action $\mathcal{L}(i) \in \mathcal{A}(i)$. Let $\mathcal{E}$ be the finite policy space with $|\mathcal{E}| = \prod_{i=1}^{|\mathcal{S}|} |\mathcal{A}(i)|$. For a given policy $\mathcal{L}$, the Markov chain evolves according to the transition matrix $P^{\mathcal{L}}$, where $\left[P^{\mathcal{L}}\right]_{i,j} = p^{\mathcal{L}(i)}(i,j)$, and the column vector of cost functions for each state-action pair is given by $f^{\mathcal{L}} = [f(1, \mathcal{L}(1)), f(2, \mathcal{L}(2)), \cdots, f(|\mathcal{S}|, \mathcal{L}(|\mathcal{S}|))]^T$. We assume $P^{\mathcal{L}}$ and $f^{\mathcal{L}}$ are unknown for all $\mathcal{L} \in \mathcal{E}$.

Assumption 1 *There exist a known $\epsilon > 0$ and a known integer $K > 0$ such that for all $\mathcal{L} \in \mathcal{E}$ and $k \geq K$, $\left[(P^{\mathcal{L}})^K\right]_{i,j} > \epsilon$ for all $i, j = 1, \ldots, |\mathcal{S}|$.*

Remark 1 *For simplicity of notation, we assume throughout the paper that $K = 1$. When $K > 1$, the results still apply by introducing a* dwell time *of at least K time steps in the adaptive optimization scheme. That is, the scheme must stay with a given policy for the dwell time before updating the estimate of the O-potential or introducing an experimental (forcing) action. Consequently, assuming $K = 1$ introduces no loss of generality.*

Assumption 1 implies the Markov chain is ergodic under any policy. We denote the steady-state probabilities corresponding to the Markov chain running under policy $\mathcal{L}$ by a row vector $\pi^{\mathcal{L}}$. The long-run average cost of Markov chain under policy $\mathcal{L}$ is then

$$\eta^{\mathcal{L}} = \lim_{N\to\infty} \frac{1}{N+1} \sum_{n=0}^{N} E^{X_0,\mathcal{L}}\{f(X_n, \mathcal{L}(X_n))\} = \pi^{\mathcal{L}} f^{\mathcal{L}},$$

where $E^{X_0,\mathcal{L}}\{\cdot\}$ denotes the expectation determined by the initial state X_0 of the Markov chain running under policy $\mathcal{L}$. The optimization objective is to find

$$\mathcal{L}^* = \arg\min_{\mathcal{L}\in\mathcal{E}} \eta^{\mathcal{L}}.$$

When there are multiple optimal policies, we will use $\mathcal{L}^*$ to denote a particular optimal policy.

Formally, an *adaptive optimization rule* Υ is a series of action-valued random variables $\Upsilon = \{U_n\}_{n=0}^{\infty}$ such that, at each decision time n,

$$\{U_n = u \in \mathcal{A}(X_n))\} \in \sigma\{X_0, U_0, F_0 \cdots, X_{n-1}, U_{n-1}, F_{n-1}, X_n\},$$

where $\sigma\{\cdots\}$ denotes the σ-field generated by the random variables, and F_n is an observed random cost with

$$E\{F_n \mid \mathcal{F}_n \} = f(X_n, U_n),$$

where σ-field

$$\mathcal{F}_n = \sigma\{X_0, U_0, F_0, \cdots, X_{n-1}, U_{n-1}, F_{n-1}, X_n, U_n\}$$

with

$$\mathcal{F}_0 = \sigma\{X_0, U_0\}.$$

Assumption 2 *There exists a finite $M > 0$ such that*

$$E\{F_n^2 \mid \mathcal{F}_n\} < M.$$

Given an adaptive optimization rule Υ, we define an associated sequence of polices $\{\mathcal{L}_n\}_{n=0}^{\infty}$ recursively for a given sample path as follows. The initial policy is given by $\mathcal{L}_0(X_0) = U_0$ and subsequent policies for $n \geq 1$ are given by $\mathcal{L}_n(X_n) = U_n$ and $\mathcal{L}_n(i) = \mathcal{L}_{n-1}(i)$ for $i \neq X_n$.

Our objective is to find an optimization rule Υ such that

$$\lim_{N\to\infty} \frac{1}{N+1} \sum_{n=0}^{N} E^{X_0,\Upsilon}\{f(X_n, U_n)\} = \eta^{\mathcal{L}^*}, \tag{1}$$

and

$$\lim_{N\to\infty} \frac{1}{N+1} \sum_{n=0}^{N} 1_{\{\mathcal{L}_n = \mathcal{L}^*\}} = 1, \ w.p.1 \ [\mathcal{P}^{X_0,\Upsilon}], \tag{2}$$

where $E^{X_0,\Upsilon}$ and $\mathcal{P}^{X_0,\Upsilon}$ denote respectively the expectation and probability measure determined by the initial state X_0 of the Markov chain running the adaptive optimization rule Υ. Thus, we are interested in designing an adaptive rule which achieves the optimal long-run average cost and at the same time converges in a "Cesaro" [6] sense.

3 Performance Potentials

In this section, we review the concept of performance potentials for the standard MDP problem with long-run average cost. (See [4] for the details of results described in this section.)

It can be proved that fundamental matrix $I - P^{\mathcal{L}} + e\pi^{\mathcal{L}}$ is invertible for any ergodic $\mathcal{L}$, where $e = [1, 1, \cdots, 1]^T$. For any policy $\mathcal{L} \in \mathcal{E}$, if $\tilde{g}^{\mathcal{L}}$ is defined by

$$\tilde{g}^{\mathcal{L}} = (I - P^{\mathcal{L}} + e\pi^{\mathcal{L}})^{-1} f^{\mathcal{L}}, \tag{3}$$

it follows directly from (3) that

$$\pi^{\mathcal{L}} \tilde{g}^{\mathcal{L}} = \eta^{\mathcal{L}}.$$

The *performance potential vector* for policy $\mathcal{L}$ is given by

$$g^{\mathcal{L}} = \tilde{g}^{\mathcal{L}} + \rho e \tag{4}$$

for an arbitrary constant ρ. The components of $g^{\mathcal{L}}$ are called *performance potentials.*

The long-run average performance of the Markov chain for any two policies $\mathcal{L}$ and $\mathcal{L}'$ can be compared using the performance potential vector for one of the policies, $g^{\mathcal{L}}$, according to

$$\eta^{\mathcal{L}'} - \eta^{\mathcal{L}} = \pi^{\mathcal{L}'}[(P^{\mathcal{L}'} g^{\mathcal{L}} + f^{\mathcal{L}'}) - (P^{\mathcal{L}} g^{\mathcal{L}} + f^{\mathcal{L}})]. \tag{5}$$

If the performance potential vector $g^{\mathcal{L}^*}$ for the optimal policy $\mathcal{L}^*$ is known, it follows from (5) that the optimal policy is given for each $i = 1, \ldots, |\mathcal{S}|$ by

$$\mathcal{L}^*(i) = \arg \min_{\alpha \in \mathcal{A}(i)} \left\{ \sum_{j=1}^{|\mathcal{S}|} p^{\alpha}(i,j) g^{\mathcal{L}^*}(j) + f(i,\alpha) \right\}. \tag{6}$$

Since the expression in (6) for each state i is independent of the expressions for the other states, the optimal policy for all states can be written in one vector-valued equation as

$$\mathcal{L}^* = \arg \min_{\mathcal{L} \in \mathcal{E}} \left\{ P^{\mathcal{L}} g^{\mathcal{L}^*} + f^{\mathcal{L}} \right\}, \tag{7}$$

where the minimum is taken for each component of the vector.

4 O-potentials

We define the *optimal-direction potentials*, or simply *O-potentials*, for each $i \in \mathcal{S}$ and $\alpha \in \mathcal{A}(i)$ as

$$c(i,\alpha) = \sum_{j=1}^{|\mathcal{S}|} p^{\alpha}(i,j) g^{\mathcal{L}^*}(j) + f(i,\alpha). \tag{8}$$

The optimal policy (6) can then be written as

$$\mathcal{L}^*(i) = \arg \min_{\alpha \in \mathcal{A}(i)} c(i,\alpha) \quad i = 1, 2, \cdots, |\mathcal{S}|.$$

For a given policy $\mathcal{L} \in \mathcal{E}$, we define the *optimal-direction potential vector for* $\mathcal{L}$, or simply the *O-potential vector for* $\mathcal{L}$, by

$$c^{\mathcal{L}} = P^{\mathcal{L}} g^{\mathcal{L}^*} + f^{\mathcal{L}}. \tag{9}$$

That is, $c^{\mathcal{L}}(i) = c(i, \mathcal{L}(i))$ for $i = 1, \ldots, |\mathcal{S}|$.

The motivation for the term *O-potential* can be seen from

$$\eta^{\mathcal{L}} - \eta^{\mathcal{L}^*} = \pi^{\mathcal{L}}(c^{\mathcal{L}} - c^{\mathcal{L}^*}), \tag{10}$$

which follows from (5) and (9). In words, (10) states that the deviation of the performance of any policy $\mathcal{L}$ from the performance of the optimal policy $\mathcal{L}^*$ is a function of the direction of the deviation of $c^{\mathcal{L}}$, the O-potential vector for $\mathcal{L}$, from $c^{\mathcal{L}^*}$ the O-potential vector for $\mathcal{L}^*$. From (7), we have that

$$c^{\mathcal{L}} \geq c^{\mathcal{L}^*} \text{ for all } \mathcal{L} \in \mathcal{E},$$

where equality is achieved only for optimal policies. It can be verified from (3) and (4) that

$$g^{\mathcal{L}^*} = P^{\mathcal{L}^*} g^{\mathcal{L}^*} + f^{\mathcal{L}^*} - \eta^{\mathcal{L}^*} e = c^{\mathcal{L}^*} - \eta^{\mathcal{L}^*} e, \tag{11}$$

which shows that the O-potential at the optimal policy is a version of performance potential for the optimal policy.

Returning to the adaptive optimization problem, our objective is to find a recursive scheme for estimating the O-potential $c(i, \alpha)$ (8) and to use the estimates to achieve an optimal policy $\mathcal{L}^*$. To derive an equation for the O-potential that can be used as the basis for a recursive estimation scheme, we first note that (9) and (11) imply

$$c^{\mathcal{L}} = P^{\mathcal{L}}(c^{\mathcal{L}^*} - e\eta^{\mathcal{L}^*}) + f^{\mathcal{L}} = P^{\mathcal{L}} \min_{\mathcal{L}' \in \mathcal{E}} c^{\mathcal{L}'} + f^{\mathcal{L}} - e\eta^{\mathcal{L}^*}. \tag{12}$$

Since knowing $c^{\mathcal{L}}$ for all possible policies is equivalent to knowing $c(i, \alpha)$ for all state-action pairs, (12) implies

$$c(i, \alpha) = \sum_{j=1}^{|\mathcal{S}|} p^{\alpha}(i, j) \min_{\alpha' \in \mathcal{A}(j)} c(j, \alpha') + f(i, \alpha) - \eta^{\mathcal{L}^*}. \tag{13}$$

Lemma 1 *If c is the solution to (13), and define policy $\bar{\mathcal{L}}$ with*

$$\bar{\mathcal{L}}(i) = \arg \min_{\alpha \in \mathcal{A}(i)} c(i, \alpha),$$

we have $\eta^{\bar{\mathcal{L}}} = \eta^{\mathcal{L}^*}$, i.e., $\bar{\mathcal{L}}$ is an optimal policy.

Proof: Equation (13) implies that for any policy $\mathcal{L}$,

$$c^{\mathcal{L}} = P^{\mathcal{L}} c^{\bar{\mathcal{L}}} + f^{\mathcal{L}} - e\eta^{\mathcal{L}^*}. \tag{14}$$

Multiplying both sides of (14) with $\pi^{\mathcal{L}}$, we have

$$\pi^{\mathcal{L}}(c^{\mathcal{L}} - c^{\bar{\mathcal{L}}}) = \eta^{\mathcal{L}} - \eta^{\mathcal{L}^*} \geq 0. \tag{15}$$

Since (15) must hold for any $\mathcal{L}$, it follows that $\eta^{\bar{\mathcal{L}}} = \eta^{\mathcal{L}^*}$. □

Although Lemma 1 states that a solution to (13) is necessarily a version of O-potentials, we do not use this equation for our recursive estimation procedure. The reason is two-fold: first, the solution to (13) is not unique; secondly, the term $\eta^{\mathcal{L}^*}$ is unknown. To circumvent both of these problems, we modify (13) slightly to obtain the following equation for the O-potential:

$$c(i, \alpha) = \sum_{j=1}^{|\mathcal{S}|} p^{\alpha}(i, j) \min_{\alpha' \in \mathcal{A}(j)} c(j, \alpha') + f(i, \alpha) - \sum_{k=1}^{|\mathcal{S}|} \mu(k) \min_{\alpha'' \in \mathcal{A}(k)} c(k, \alpha''), \tag{16}$$

or in a policy-based form,

$$c^{\mathcal{L}} = (P^{\mathcal{L}} - e\mu) \min_{\mathcal{L}' \in \mathcal{E}} c^{\mathcal{L}'} + f^{\mathcal{L}}, \tag{17}$$

where row vector μ should satisfy

$$||P^{\mathcal{L}} - e\mu|| < 1 \text{ for all } \mathcal{L} \in \mathcal{E}. \tag{18}$$

Remark 2 *By Assumption 1, there always exists a row vector μ that satisfies (18) (for $K = 1$). In particular, we assume either $\mu = \delta e^T$ or $\mu = \delta e_k^T$, where δ satisfies $0 < \delta \leq \epsilon$ and e_k is the kth column of the identity matrix for some $k \in \{1, \ldots, |S|\}$.*

The following lemma shows the feasibility of using (16) for the estimation of O-potentials.

Lemma 2 *If c is a solution to (16), it is also a solution to (13) corresponding to ρ in (4) given by*

$$\rho = \frac{\eta^{\mathcal{L}^*} - \sum_{i=1}^{|S|} \mu(i)(P^{\mathcal{L}^*} \bar{g}^{\mathcal{L}^*})(i) - \sum_{i=1}^{|S|} \mu(i) f^{\mathcal{L}^*}(i)}{\sum_{i=1}^{|S|} \mu(i)}.$$

Moreover, if the row vector μ satisfies (18), a solution to (16) exists and is unique.

Proof: The first statement follows from direct calculation. $||P^{\mathcal{L}^*} - e\mu|| < 1$, implies $I - P^{\mathcal{L}^*} + e\mu$ is invertible. Therefore, there exists a unique solution x^* to the equation

$$(I - P^{\mathcal{L}^*} + e\mu)x = f^{\mathcal{L}^*}. \tag{19}$$

But this implies

$$x^* = (P^{\mathcal{L}^*} - e\mu)x^* + f^{\mathcal{L}^*}. \tag{20}$$

Comparing (20) with (17), we see that $x^* = c^{\mathcal{L}^*}$. Moreover, from $\mathcal{L}^*$ one can compute $c(i, \alpha)$ from (16) for all state-action pairs. Therefore, the solution to (16) exists and is unique. □

5 Adaptive Optimization

Our recursive estimation scheme for the O-potentials is based on stochastic approximation. Corresponding to (16), the recursive update of O-potentials at time $n + 1$ for all feasible pairs (i, α) is

$$\begin{aligned} c_{n+1}(i, \alpha) &= (1 - \gamma_{n+1,i,\alpha})c_n(i, \alpha) \\ &+ \gamma_{n+1,i,\alpha}[\min_{\alpha' \in \mathcal{A}(X_{n+1})} c_n(X_{n+1}, \alpha') + F_n - \sum_{k=1}^{|S|} \mu(k) \min_{\alpha'' \in \mathcal{A}(k)} c_n(k, \alpha'')], \end{aligned} \tag{21}$$

and the step-size should satisfy the following conditions.

Condition 1 *For all i and $\alpha \in \mathcal{A}(i)$, $c(i, \alpha)$ is updated at an infinite set of times $N^{i,\alpha} = \{n + 1 : X_n = i, U_n = \alpha\}$ along the sample path, and $\gamma_{n,i,\alpha} = 0$ for $n \notin N^{i,\alpha}$.*

Condition 2 $\gamma_{n,i,\alpha} \geq 0$, $\sum_{n \in N^{i,\alpha}} \gamma_{n,i,\alpha} = \infty$, *and* $\sum_{n \in N^{i,\alpha}} \gamma_{n,i,\alpha}^2 < \infty$.

Our adaptive optimization scheme is defined as follows. We first choose any increasing sequence $T = \{t_i\}_{i=1}^{\infty}$ of positive integers, such that

$$\lim_{N\to\infty} \frac{1}{N+1} \sum_{n=0}^{N} 1_{\{n\in T\}} = 0. \tag{22}$$

For example, we can set $t_i = [(i-1)^{\frac{1}{\zeta}}]$ with some $\zeta < 1$, where $[\cdot]$ is the integer function.

Adaptive Optimization Rule Υ
At each time n, estimate c according to (21), where the step-size is given by

$$\gamma_{n+1,X_n,U_n} = \frac{1}{|\{j \le n+1 \mid j \in N^{X_n,U_n}\}|^{\beta}}, \quad 0.5 < \beta \le 1, \tag{23}$$

and we apply actions according to

$$U_n = \begin{cases} \alpha \in \mathcal{A}(X_n) \ \text{ with probability} \frac{1}{|\mathcal{A}(X_n)|} & n \in T; \quad \text{(experiment time)} \\ \arg\min_{\alpha\in\mathcal{A}(X_n)} c_n(X_n,\alpha) & n \notin T, \quad \text{(decision time)} \end{cases} \tag{24}$$

In words, we choose the actions based on an estimate c_n of c for $n \notin T$, and apply experimental actions (*forcing*) for $n \in T$. The randomized forcing scheme used here can be replaced with other schemes, such as sequential forcing where actions are applied to the system sequentially at experiment times. At decision time n, in case that there are multiple actions for $\arg\min_{\alpha\in\mathcal{A}(X_n)} c_n(X_n,\alpha)$, let $U_n = \mathcal{L}_{n-1}(X_n)$, if $\mathcal{L}_{n-1}(X_n)$ is one of those actions; otherwise let U_n be any one of the options.

To prove that the above optimization scheme achieves objectives (1) and (2), we start with the following boundedness lemma.

Lemma 3 *If the step-size satisfies Conditions 1 and 2, the sequence of functions c_n generated by (21) is bounded with probability one.*

Proof: Rewrite (21) as

$$c_{n+1}(i,\alpha) = (1-\gamma_{n+1,i,\alpha})c_n(i,\alpha) + \gamma_{n+1,i,\alpha}\Delta_n$$
$$+\gamma_{n+1,i,\alpha}[E\{\min_{\alpha'\in\mathcal{A}(X_{n+1})} c_n(X_{n+1},\alpha') \mid \mathcal{F}_n\} + f(X_n,U_n) - \sum_{k=1}^{|S|}\mu(k)\min_{\alpha''\in\mathcal{A}(k)} c_n(k,\alpha'')].$$

where

$$\Delta_n = \min_{\alpha'\in\mathcal{A}(X_{n+1})} c_n(X_{n+1},\alpha') - E\{\min_{\alpha'\in\mathcal{A}(X_{n+1})} c_n(X_{n+1},\alpha') \mid \mathcal{F}_n\} + F_n - f(X_n,U_n).$$

We have

$$E\{\Delta_n \mid \mathcal{F}_n\} = 0$$

and, from Assumption 2,

$$E\{\Delta_n^2 \mid \mathcal{F}_n\} \le 2\|c_n\|^2 + 2M. \tag{25}$$

From Lemma 1, let c be the unique solution to (16). We have

$$|E\{\min_{\alpha'\in\mathcal{A}(X_{n+1})} c_n(X_{n+1},\alpha') \mid \mathcal{F}_n\} + f(X_n,U_n) - \sum_{k=1}^{|S|}\mu(k)\min_{\alpha''\in\mathcal{A}(k)} c_n(k,\alpha'')$$
$$-E\{\min_{\alpha'\in\mathcal{A}(X_{n+1})} c(X_{n+1},\alpha') \mid \mathcal{F}_n\} + f(X_n,U_n) - \sum_{k=1}^{|S|}\mu(k)\min_{\alpha''\in\mathcal{A}(k)} c(k,\alpha'')|$$
$$\le m\|c_n - c\| \le m(\|c_n\| + \|c\|),$$

where $m = \max_{\mathcal{L}\in\mathcal{E}} \|P^{\mathcal{L}} - e\mu\| < 1$.

Define the non-decreasing $\mathcal{F}_n$-measurable random variable sequence B_n:

$$B_0 = \frac{1 + \|c_0\| + 2(m+1)\|c\|}{1-m} \text{ and } B_n = B_{n-1}(1+\theta)^{K_n},\ n \geq 1,$$

where $\theta = \frac{(1-m)(1+\|c_0\|)}{m[1+\|c_0\|+2(m+1)\|c\|]}$ and

$$K_n = \begin{cases} 0 & \text{if } \log_{1+\theta}\frac{\|c_n\|}{B_{n-1}} \leq 1 \\ \log_{1+\theta}\frac{\|c_n\|}{B_{n-1}} & \text{if } \log_{1+\theta}\frac{\|c_n\|}{B_{n-1}} > 1 \end{cases}.$$

From the definition, we can see that $\|c_n\| \leq (1+\theta)B_n$; and $\|c_n\| = B_n$ if $B_n > B_{n-1}$. Moreover, we have

$$|E\{\min_{\alpha'\in\mathcal{A}(X_{n+1})} c_n(X_{n+1},\alpha') \mid \mathcal{F}_n\} + f(X_n,U_n) - \sum_{k=1}^{|S|}\mu(k)\min_{\alpha''\in\mathcal{A}(k)} c_n(k,\alpha'')| \leq B_n$$

Since the step-size satisfies Condition 1 and Condition 2, there exists a finite n', such that $|\gamma_{n+1,X_n,U_n}| < 1$ for $n \geq n'$. For any $n \geq n_0 \geq 0$, all i and $\alpha \in \mathcal{A}(i)$, define

$$\widetilde{\Delta}^{i,\alpha}_{n,n_0} = (1-\gamma_{n,i,\alpha})\widetilde{\Delta}^{i,\alpha}_{n-1,n_0} + \gamma_{n,i,\alpha}\frac{\Delta_n}{B_n}$$

with $\widetilde{\Delta}^{i,\alpha}_{n_0,n_0} = 0$.

By a standard stochastic approximation result [5], we have

$$\lim_{n\to\infty}\widetilde{\Delta}^{i,\alpha}_{n,0} = 0, \quad w.p.1$$

because the martingale difference $\frac{\Delta_n}{B_n}$ has bounded conditional second-moment

$$E\{\frac{\Delta_n^2}{B_n^2} \mid \mathcal{F}_n\} \leq 2(1+\theta)^2 + 2M$$

from (25). This shows that, with probability one, there exists finite n'' such that for all $n \geq n'' \geq n'$ and all (i,α)

$$|\widetilde{\Delta}^{i,\alpha}_{n,0}| \leq \frac{\theta}{2}.$$

Let $\Xi = \{n : \|c_n\| = B_n\ ,\ n \geq n''\}$. If $\Xi = \emptyset$, then $B_n = B_{n''}$ for all $n \geq n''$ and we have boundedness automatically. Suppose $\Xi \neq \emptyset$, and let $n_1 \geq n''$ satisfy

$$\|c_{n_1}\| = B_{n_1}.$$

We can prove that for $n \geq n_1$, all i and $\alpha \in \mathcal{A}(i)$

$$(\widetilde{\Delta}^{i,\alpha}_{n,n_1} - 1)B_{n_1} \leq c_n(i,\alpha) \leq (\widetilde{\Delta}^{i,\alpha}_{n,n_1} + 1)B_{n_1} \quad n \geq n_1 \tag{26}$$

by using induction along time n.

(26) yields that, for all $n \geq n_1$, we have

$$\|c_n\| \leq (1+\theta)B_{n_1}.$$

This finishes the boundedness proof. □

With the above boundedness lemma, we have the following convergence theorem for the recursive estimation.

Theorem 1 *If the step-size satisfies Conditions 1 and 2, then for any i and $\alpha \in \mathcal{A}(i)$,*

$$\lim_{n\to\infty} c_n(i,\alpha) = c(i,\alpha), \ w.p.1. \tag{27}$$

Proof: The proof for this theorem is a continuation of the proof for the boundedness Lemma.

Define $O_k = \sqrt{m} O_{k-1}$, $k \geq 2$ with $O_1 = (1+\theta)B_{n_1} + ||c||$. We can prove that, with probability one, for all k, there exists finite m_k such that for all $n \geq m_k$,

$$||c_n - c|| \leq O_k. \tag{28}$$

Since $\lim_{k\to\infty} O_k = 0$, this will also give the proof for

$$\lim_{n\to\infty} ||c_n - c|| = 0, \ w.p.1,$$

which in turn leads to this theorem. See [11] for details. □

We can now state and prove the main result.

Theorem 2 *The adaptive optimization rule Υ defined by (23) and (24) achieves the performance objectives (1) and (2).*

Proof: We can prove that the step-size (23) satisfies Condition 1 and Condition 2 under adaptive rule Υ. From Theorem 1, we have the convergence of all O-potentials to the true values. Therefore, optimal policy will always be chosen after a certain finite starting time except at the experimental times with probability one. We only need to show the finite starting time and epertimental times (which satisfy (22)) will not affect the optimal long run average cost. See [11] for details. □

6 Example

The following small example demonstrates the adaptive optimization scheme.

Example: $\mathcal{S} = \{1,2\}$, $\mathcal{A} = \{\alpha_1, \alpha_2\}$, $\mathcal{A}(1) = \mathcal{A}(2) = \{\alpha_1, \alpha_2\}$; $p^{\alpha_1}(1,\cdot) = [0.1, 0.9]$, $p^{\alpha_1}(2,\cdot) = [0.1, 0.9]$, $p^{\alpha_2}(1,\cdot) = [0.9, 0.1]$, $p^{\alpha_2}(2,\cdot) = [0.9, 0.1]$; $f(1,\alpha_1) = 0.2$, $f(2,\alpha_1) = 0.3$, $f(1,\alpha_2) = 0.7$, $f(2,\alpha_2) = 0.8$; $\mathcal{E} = \{(\alpha_1,\alpha_1), (\alpha_1,\alpha_2), (\alpha_2,\alpha_1), (\alpha_2,\alpha_2)\}$; $\eta^{(\alpha_1,\alpha_1)} = 0.29$, $\eta^{(\alpha_1,\alpha_2)} = 0.5$, $\eta^{(\alpha_2,\alpha_1)} = 0.5$, $\eta^{(\alpha_2,\alpha_2)} = 0.71$; $\mathcal{L}^* = (\alpha_1, \alpha_1)$, $\eta^{\mathcal{L}^*} = 0.29$.

Table 1: Simulation results ($\delta = 0.05$, $\beta = 0.55$)

Simu. #	ζ	μ/δ	$c(1,\alpha_1)$ $c(1,\alpha_2)$ $c(2,\alpha_1)$ $c(2,\alpha_2)$	Estimated $c(1,\alpha_1)$ $c(1,\alpha_2)$ $c(2,\alpha_1)$ $c(2,\alpha_2)$	Sample-path-average-cost
1	0.1	[1,1]	2.85 3.27 2.95 3.37	2.85 2.96 2.95 3.03	0.29
2		[1,0]	5.80 6.22 5.90 6.32	5.80 5.82 5.90 5.94	0.29
3		[0,1]	5.70 6.12 5.80 6.22	5.70 5.72 5.80 5.81	0.29
4	0.9	[1,1]	2.85 3.27 2.95 3.37	2.85 3.27 2.95 3.37	0.36
5		[1,0]	5.80 6.22 5.90 6.32	5.80 6.22 5.90 6.32	0.36
6		[0,1]	5.70 6.12 5.80 6.22	5.70 6.12 5.80 6.22	0.36

Each simulation experiment was performed along a sample path with 20000 transitions. There is a tradeoff in selecting ζ: smaller ζ means less frequent experimental times, which may lead to insufficiently accurate O-potential estimates and thus possibly a non-optimal policy for a very long time; larger ζ means more frequent experimental times, which will lead to better estimates of the O-potentials at the expense of poorer performance over a finite time horizon. We observe that, in simulations 1, 2 and 3, optimal long-run average cost is in fact obtained though the estimates of some O-potentials are far from accurate. This is because the estimated O-potentials have the same order as the optimal policy in this case. On the other hand, in simulations 4, 5 and 6, average cost performance is sacrificed for the accuracy of estimation as expected.

References

[1] J. Abounadi, D. Bertsekas and V.S. Borkar, "Learning Algorithms for Markov Decision Processes with Average cost," Tech. Report, Lab. for Info. and Decision Systems, MIT, October 1998.

[2] D.P. Bertsekas and J.N. Tsitsiklis, *Neuro-Dynamic Programming,* Athena Scientific, Belmont, Massachusetts, 1996.

[3] V. Borkar and P. Varaiya, "Adaptive Control of Markov Chains, I: Finite Parameter Set," *IEEE Transactions on Automatic Control,* Vol. 24, pp. 953-958, 1979.

[4] X. R. Cao, "Single Sample Path Based Optimization of Markov Chains," *Journal of Optimization: Theory and Applications*, to appear, 1998.

[5] H. J. Kushner and D. S. Clark, *Stochastic Approximation Methods for Constrained and Unconstrained Systems,* Springer-Verlag, Berlin, 1978.

[6] P. R. Kumar, "A Survey of Some Results in Stochastic Adaptive Control," *SIAM Journal on Control and Optimization,* Vol. 23, No. 3, pp. 329-380, 1985.

[7] L. Ljung, "Analysis of Recursive Stochastic Algorithms," *IEEE Transactions on Automatic Control,*, Vol. 22, pp. 551-575, 1977.

[8] M. Loève, *Probability Theory,* D. Van Nostrand, Princeton, New Jersey, 1963.

[9] P. Mandl, "Estimation and Control in Markov Chains," *Advances in Applied Probability,* Vol. 6, pp. 40-60, 1974.

[10] Z. Y. Ren, *The Applications of Performance Potential to Sensitivity Analysis and Optimization on Markov Chains, MPhil. Thesis,* The Hong Kong University of Science and Technology, 1998.

[11] Z. Y. Ren and Bruce H. Krogh, "Adaptive Control of Markov Chains with Average Cost," submitted to *SIAM Journal on Control and Optimization,* April, 1999.

[12] D. J. White, "Dynamic Programming, Markov Chains, and the Method of Successive Approximations," *Journal of Mathematical Analysis and Applications,* Vol. 6, pp. 363-376, 1963.

[13] C. Watkins and P. Dayan, "Q-learning," Machine Learning 8, pp. 279-292, 1992.

Tracking Capability Analysis of the LMS Algorithm for FIR Systems with AR Coefficients *

Yongbin Wei, Saul B. Gelfand, and James V. Krogmeier

School of Electrical and Computer Engineering
Purdue University
West Lafayette, IN 47907-1285
{ywei,gelfand,jvk}@ecn.purdue.edu

Abstract

This paper proposes a criterion to classify the tracking ability of an adaptive algorithm, when it is used to identify a time-varying system. The criterion is then applied to define the trackable region where the LMS algorithm can track an FIR model with AR coefficients. Using extensions of results for LMS stability and performance appropriate for this model, an explicit expression for the trackable region is obtained. The results demonstrate certain fundamental limitations in tracking with the LMS algorithm. In particular, it is shown that there generally exist coefficient variation rates above which LMS performance is worse than simple offline estimation. Stability is incorporated directly into the trackable region analysis, and tracking performance is optimized over stable step-size to obtain the trackable region. The existence of an optimum step-size turns out to be directly related to the extent of the trackable region. Computer simulations are conducted to further understand and validate the results.

1 Introduction

The least mean square (LMS) algorithm is widely used in adaptive filtering and system identification, and has been the subject of many theoretical investigations [1]. The theoretical investigations typically involve specifying conditions for stability and characterizing the tracking performance of LMS (stability is strictly required for tracking).

In recent work, Guo *et al.* [2] have examined the exponential stability of the LMS algorithm for a large class of signal and system models, including the finite impulse response (FIR) system/channel model with autoregressive (AR) coefficients that is the primary focus of this paper. As for characterizing the tracking performance of the LMS algorithm, many results (and methods) appear in the literature. Guo *et al.* [2] give bounds and estimates for the error variance, which depend on the step-size and the model parameters. However, due to the fact that the results were based on some very

---

[*] This work was partially supported by the National Science Foundation under Grant NCR-9706372, and US Army Research Office under Grant DAAG55-98-1-0360, and a Purdue Research Foundation Fellowship.

mild assumptions, and applicable to a large class of models, no exact expressions for the error variance were provided. Solo [3] also gives estimates for the error variance, and considers in detail the effect of speed of adaptation (determined by the step-size) relative to speed of system variation. Perturbation techniques are applied there to obtain expansions for the error covariance, in terms of step-size and model parameters. In [2, 3] the focus is on "small" step-size and "slowly-varying" systems (the latter is related to the prior when system parameters are fixed), which are necessary to make the perturbation analysis tractable and/or accurate. However, it is not evident from these analyses how small the step-size or how slowly varying the system must be (this also makes optimization over the step-size to obtain the "best" performance questionable). Clearly, the small step-size and/or slowly varying system assumptions can be violated in practice, and it is thus of importance to understand the tracking performance of the LMS algorithm with large step-size and/or fast varying systems, including the important issue of how and when performance degrades as one moves from slow to fast variations.

Without a significant restriction on the system model, it seems technically intractable to carry out such an analysis for large step-size and/or fast varying systems. The difficulty can be circumvented by assuming a simple model of the system variation, such as bounded-variation or random-walk [4]. Solo and Kong [5] have considered the case with AR coefficients. The AR model is mathematically tractable and is appropriate for many applications, such as wireless communications where it is used to model frequency selective fading channel coefficients, where the speed of system variations is tied to the fading rate for the channel [6, 7].

Related to the question of evaluating the performance of LMS with fast system variations, is an assessment of how fast a variation rate can be tracked. The interest here is in understanding whether there is a simple and useful way by which trackable and untrackable variations can be distinguished. This topic of trackability seems less investigated. A criterion is proposed in [4] based on the *non-stationarity degree* to determine whether an adaptive filter is useful for tracking. Although simple and potentially useful, we believe there are certain inadequacies associated with the method (this is discussed in [8]). It seems reasonable that other approaches which address the trackability problem should be investigated to further understand the useful limits of the LMS algorithm.

The contributions of this paper are two-fold: first, a criterion is proposed to quantitatively answer whether an adaptive algorithm is useful for tracking a time-varying system. Second, an explicit trackability criterion for LMS tracking AR coefficient variations is derived. The derivation uses the performance of LMS in tracking FIR systems with AR coefficients (without using small step-size or slow varying assumptions), which is a generalization of the results from [5] and presented with details in [8].

The paper is organized as follows. In Section 2, we set up the system model and propose a general trackability criterion. We then define the trackable region where LMS can track AR coefficient variations. An explicit formula specifying this trackable region is derived and interpreted in Section 3. Section 4 reports simulation results.

2 The Set-up

2.1 System Model

Consider the following time-varying FIR system model,

$$y_k = \mathrm{c}_k^T \mathrm{x}_k + n_k, \qquad \text{for } k \geq 0, \tag{1}$$

where $\{\mathbf{x}_k\}$ is an N−dimensional i.i.d. Gaussian vector sequence with zero-mean and covariance $\mathbf{R}_x = \text{diag}\{\lambda_1, \lambda_2, \cdots, \lambda_N\}$, $\{n_k\}$ is an additive white Gaussian noise process with zero-mean and variance σ_n^2, and $\{\mathbf{x}_k\}$ and $\{n_k\}$ are independent of each other. The channel $\{\mathbf{c}_k\}$ is time-varying and satisfies a Gaussian AR model

$$\mathbf{c}_{k+1} = (\mathbf{I} - \mathbf{A})\mathbf{c}_k + \delta_k, \qquad \text{for } k \geq 0, \tag{2}$$

where $\{\delta_k\}$ is an i.i.d. Gaussian vector sequence with mean $\mathbf{0}$ and covariance $\mathbf{D} = \text{diag}\{\rho_1, \cdots, \rho_N\}$, and $\{\delta_k\}$ is independent of $\{\mathbf{x}_k\}$ and $\{n_k\}$. Here we assume $\mathbf{A} = \text{diag}\{a_1, a_2, \cdots, a_N\}$ with $a_i \in (0,1)$ for $1 \leq i \leq N$. In contrast to the "slow varying" assumption[1], the a_i's may take any value in $(0,1)$, and an a_i close to 1 corresponds to an extreme fast system variation. Since we are interested in "steady-state" tracking, we focus on the asymptotic regime and assume that process statistics and distributions have converged to their steady-state values.

We examine a problem where the above system is identified by a linear adaptive algorithm, as shown in Fig. 1, where $\hat{y}_k = \mathbf{w}_k^T \mathbf{x}_k$ is a linear estimate of y_k based on $\mathbf{x}_k$, and (in view of the system identification model) $\mathbf{w}_k$ is an estimate of $\mathbf{c}_k$. The above assumptions, which correspond to the so-called independent Gaussian data case [5], are structured enough to give concrete results, as will be seen in the sequel. It is often assumed in the adaptive filtering literature that results based on this case hold approximately under more general conditions.

2.2 Criterion of Trackability

Our approach to specify a criterion for trackability of a time-varying (stochastic) system model with an adaptive algorithm, is to compare the performance of the adaptive estimator it generates with that of a simple estimator, and assert that the adaptive algorithm can track if and only if the performance of the adaptive estimator is better. Here the performance of the adaptive estimate is taken at the optimal adaptive algorithm settings (e.g., the step-size which optimizes performance). It is seen that our trackability criterion may give an optimistic view of when the adaptive algorithm can track (and a pessimistic view of when it cannot), especially if the performance of the simple estimator is poor. However, when an adaptive estimator has worse performance than the simple estimator for some nontrivial system model parameters, this goes a long way towards ruling out overly poor performance of the simple estimator. This will be seen to be the case below.

We first need to specify an error measure for the trackability criterion. Let $\hat{y}_k$ denote an estimate of y_k, which may be either the adaptive estimate or the simple estimate. One error measure to consider is the *Mean Square Error* (MSE), MSE $\triangleq \mathbb{E}[\,|y_k - \hat{y}_k|^2\,]$. Many adaptive algorithms (including LMS) are "derived" by minimizing MSE. Another error measure which might be considered is the *Noiseless Mean Square Error* (NMSE), NMSE $\triangleq \mathbb{E}[\,|\overline{y}_k - \hat{y}_k|^2\,]$ where $\overline{y}_k = \mathbf{c}_k^T \mathbf{x}_k = y_k - n_k$. Note that for the standard LMS algorithm, NMSE coincides with the *Excess Mean Square Error* (EMSE), since EMSE $= \text{MSE} - \sigma_n^2$. Depending upon the algorithms and estimators that are compared, MSE and NMSE can yield different criterion for trackability.

Next, we need to specify simple estimates which can be compared with the adaptive estimate, under the error measures. As a guide, it seems reasonable that the simple estimate should be based at most on the same data subset as the adaptive estimate

---

[1] Roughly speaking, the assumption requires a_i to be very "small", but it also involves σ_n^2 and $\text{tr}\,\mathbf{R}_x$ [5, page 127].

(otherwise, under nominal conditions, the performance of the simple estimate would likely be better than the adaptive estimate, and the system would be routinely labelled untrackable). General estimates $\hat{y}_k$ can be divided into prior or posterior estimates corresponding to whether they depend on the data $\{\mathbf{x}_{i+1}, y_i : i \leq k-1\}$ or $\{\mathbf{x}_i, y_i : i \leq k\}$, respectively. Linear estimates $\hat{y}_k = \mathbf{w}_k^T \mathbf{x}_k$ are hence divided into prior or posterior estimators, corresponding to whether the weight $\mathbf{w}_k$ depends on the data $\{\mathbf{x}_i, y_i : i \leq k-1\}$ or $\{\mathbf{x}_i, y_i : i \leq k\}$, respectively [2]. For adaptive algorithms which generate prior estimates (like the conventional LMS algorithm), a simple prior estimate for comparison is $\hat{y}_k = 0$. For adaptive algorithms which generate posterior estimates (like the posterior LMS algorithm [1]), a simple posterior estimate for comparison is $\hat{y}_k = y_k$. Note that for prior estimates, NMSE = MSE$-\,\sigma_n^2$, so that comparisons under NMSE and MSE are equivalent. However, for posterior estimates, NMSE $\neq$ MSE $-\,\sigma_n^2$ in general, so that comparisons under NMSE and MSE are *not* equivalent. Moreover, comparisons with the simple posterior estimate $\hat{y}_k = y_k$ under MSE are not useful, since the MSE of this estimate is 0, and consequently the time-varying system would never be classified trackable for any adaptive algorithm.

The Macchi trackability criterion [4] can be interpreted in this framework as comparing the LMS estimate with simple posterior estimate $\hat{y}_k = y_k$ under the NMSE measure. Note that the estimates are based on different data subsets, violating the above guideline. It can be shown that this leads to some counterintuitive behavior, e.g., the trackability criterion in [4] is harder to satisfy as measurement noise variance decreases. Here, we propose a trackability criterion for the LMS algorithm based on comparing the LMS estimate with the simple prior estimate $\hat{y}_k = 0$ under the MSE error measure (equivalently the NMSE error measure, since both estimates are prior estimates). The resulting criterion can be stated as follows:

$$\textit{Adaptive algorithm can track the system if and only if } \min_{\mathcal{P}} \text{MSE} < \mathbb{E}\left[\, y_k^2 \,\right], \tag{3}$$

or equivalently, *if and only if* $\min_{\mathcal{P}} \text{NMSE} < \mathbb{E}[\,\overline{y}_k^2\,]$, where $\mathcal{P}$ is the collection of the algorithm parameters (e.g., step-size). Note that the optimal performance of the algorithm is used to characterize its tracking capability, as indicated by the minimization over the algorithm parameters $\mathcal{P}$. In the above criteria, $\mathbb{E}[\,y_k^2\,]$ and $\mathbb{E}[\,\overline{y}_k^2\,]$ are equal to their steady-state values. We remark that the above criterion is potentially applicable to different adaptive algorithms and/or time-varying models. In the sequel, we focus on the LMS algorithm for tracking a time-varying FIR system with AR coefficient variations, develop an explicit characterization of the trackability criterion (3), and discuss its qualitative behavior.

2.3 Trackable Region of FIR systems with AR Coefficients by the LMS Algorithm

In the following, we consider the special case of tracking an FIR system (1) with AR coefficient variations (2), using the LMS algorithm

$$\mathbf{w}_{k+1} = \mathbf{w}_k + \alpha e_k \mathbf{x}_k \qquad \text{for} \quad k \geq 0 \quad \text{with some } \mathbf{w}_0, \tag{4}$$

where $e_k = y_k - \mathbf{w}_k^T \mathbf{x}_k$ is the estimation error, and $\alpha > 0$ is a step-size (the only adjustable parameter of LMS). Since $\mathbf{c}_k$ has mean zero, a reasonable choice for $\mathbf{w}_0$ is $\mathbf{w}_0 = 0$; in this case, the simple prior estimator $\hat{y}_k = 0$ coincides with a scheme of no adaptation (i.e.,

---

[2] Adaptively generated linear estimates typically have weights with one of these data dependencies.

$\alpha = 0$). It will be convenient to define $\mathbf{v}_k = \mathbf{w}_k - \mathbf{c}_k$, the deviation of the estimate from the true system. From the assumptions of the model, we have $\mathbb{E}[\,\overline{y}_k^2\,] = \sum_{i=1}^{N} \rho_i\lambda_i/(2a_i - a_i^2)$. Also, the NMSE and EMSE of LMS algorithm coincide. Using these results in the trackability criterion in (3), we are now ready to define the trackable region:

Definition 1 *The **trackable region** $\Re$ of AR parameters with the LMS algorithm is*

$$\Re = \left\{ (a_1, a_2, \cdots, a_N) \in (0,1)^N \;\middle|\; \inf_{\alpha \in \mathcal{S}} \mathcal{E}^{ex}(\alpha, \mathbf{R}_x, \mathbf{D}, \mathbf{A}, \sigma_n^2) < \sum_{i=1}^{N} \frac{\rho_i \lambda_i}{2a_i - a_i^2} \right\} \tag{5}$$

where $\mathcal{E}^{ex}$ is the steady state excess MSE, and $\mathcal{S}$ is the mean-square stability region:

$$\mathcal{S} = \left\{ \alpha \;\middle|\; \sum_{i=1}^{N} \frac{\alpha\lambda_i}{2 - 2\alpha\lambda_i} < 1, \;\; and \;\; 0 < \alpha\lambda_i < 1 \; for \;\; i = 1,\, 2,\, \ldots,\, N \right\}. \tag{6}$$

REMARK: We choose to focus on the AR parameters $\mathbf{A} = \mathrm{diag}\{a_1, \ldots, a_N\}$ in the definition of the trackable region $\Re$, and suppress the dependence of $\Re$ on $\mathbf{R}_x, \mathbf{D}$ and σ_n^2, for notational convenience. Note, however, that the result can be viewed in different ways, emphasizing different parameters. Note, also, that to obtain the trackable region (5), the MSE of LMS is optimized over the values of the step-size α which lie in the stable region $\mathcal{S}$. We make this optimization explicit to emphasize that only stable behavior can yield bounded MSE, a point not usually considered in step-size optimization with random walk models [3].

3 Trackable Region of FIR Systems with AR Coefficients by the LMS Algorithm

In this section, we will derive an explicit description of the trackable region $\Re$ given in (5), using the stability and performance results in [8]. We have the following

Theorem 1 *The **trackable region** $\Re$ of AR parameters with the LMS algorithm is*

$$\Re = \left\{ (a_1, a_2, \cdots, a_N) \in (0,1)^N \;\middle|\; \sum_{i=1}^{N} \frac{\lambda_i \left(a_i^3 - 2a_i^2 - \frac{(\mathrm{tr}\,\mathbf{R}_x + 4\lambda_i)\rho_i}{\sigma_n^2} a_i + \frac{2\lambda_i\rho_i}{\sigma_n^2} \right)}{(2 - a_i)a_i^2} > 0 \right\} \tag{7}$$

PROOF: In view of (5), we need to find $\inf_{\alpha\in\mathcal{S}} \mathcal{E}^{ex}(\alpha, \mathbf{R}_x, \mathbf{D}, \mathbf{A}, \sigma_n^2)$, and then compare it with $\sum_{i=1}^{N} \rho_i\lambda_i/\,(2a_i - a_i^2)$.

It is shown in [8] that, when α is in the stability region $\mathcal{S}$ given in (6), the LMS algorithm converges in mean square sense, and the limiting (steady-state) excess MSE is given by

$$\mathcal{E}^{ex}(\alpha, \mathbf{R}_x, \mathbf{D}, \mathbf{A}, \sigma_n^2) = \frac{1}{g(\alpha)} \sum_{i=1}^{N} \frac{\alpha\lambda_i}{2 - 2\alpha\lambda_i}\sigma_n^2 + \frac{1}{g(\alpha)} \sum_{i=1}^{N} \frac{\rho_i\lambda_i}{(2 - a_i)(a_i + \alpha\lambda_i(1 - a_i))(1 - \alpha\lambda_i)}, \tag{8}$$

where $g(\alpha) = 1 - \sum_{i=1}^{N} \alpha\lambda_i/\,(2 - 2\alpha\lambda_i)$.

---

[3] The mean-square stability region for random walk models is still unknown in the literature.

Let $\lambda_{\max} = \max\{\lambda_1, \ldots, \lambda_N\}$. Then some simple algebra shows that $\mathcal{S} = (0, \alpha_{\max})$, where $\alpha_{\max}$ is the unique solution of $g(\alpha) = 0$ with $\alpha \in (0, 1/\lambda_{\max})$. Hence, since $\mathcal{E}^{ex}$ is a smooth function of α, the infimum of $\mathcal{E}^{ex}$ on $\mathcal{S}$ is obtained either at a critical value of $\mathcal{E}^{ex}$ in $\mathcal{S}$, or in the limit as one of the two boundary points of $\mathcal{S}$ (i.e., 0 or $\alpha_{\max}$) is approached.

Using (8), we have

$$\frac{\partial \mathcal{E}^{ex}}{\partial \alpha} = \frac{1}{g^2(\alpha)} f(\alpha), \tag{9}$$

$$\begin{aligned} \text{where } f(\alpha) &= 2\sigma_n^2 h(\alpha) + 2h(\alpha) \sum_{i=1}^{N} \frac{\rho_i \lambda_i}{(2-a_i)(a_i + \alpha\lambda_i(1-a_i))(1-\alpha\lambda_i)} \\ &- g(\alpha) \sum_{i=1}^{N} \frac{\lambda_i^2 \rho_i \left((1-2a_i) - 2\alpha\lambda_i(1-a_i)\right)}{(2-a_i)(a_i+\alpha\lambda_i(1-a_i))^2(1-\alpha\lambda_i)^2}, \end{aligned} \tag{10}$$

and $h(\alpha) = \sum_{i=1}^{N} \lambda_i / (2 - 2\alpha\lambda_i)^2$. Since $0 < g(\alpha) < 1$ for $\alpha \in \mathcal{S}$, the critical values of $\mathcal{E}^{ex}$ in $\mathcal{S}$ and the roots of $f(\alpha)$ in $\mathcal{S}$ coincide. Now it is easy to show that $f'(\alpha) > 0$ for $\alpha \in \mathcal{S}$, so that $f(\cdot)$ is monotonically strictly increasing (and continuous) on $\mathcal{S}$. Also, we have

$$f(\alpha_{\max}) = 2\sigma_n^2 h(\alpha_{\max}) + 2h(\alpha_{\max}) \sum_{i=1}^{N} \frac{\rho_i \lambda_i}{(2-a_i)(a_i + \alpha_{\max}\lambda_i(1-a_i))(1-\alpha_{\max}\lambda_i)},$$

so $f(\alpha_{\max}) > 0$. Hence $f(\cdot)$ has a root (and $\mathcal{E}^{ex}$ has a critical point) in $\mathcal{S}$ if and only if $f(0) < 0$, and if a root exists, then it is unique. From the definition of $f(\alpha)$ in (10), $f(0) < 0$ if and only if

$$\sum_{i=1}^{N} \frac{\lambda_i \left(a_i^3 - 2a_i^2 - \frac{(\operatorname{tr} \mathbf{R}_x + 4\lambda_i)\rho_i}{\sigma_n^2} a_i + \frac{2\lambda_i \rho_i}{\sigma_n^2} \right)}{(2-a_i)a_i^2} > 0. \tag{11}$$

We consider cases below.

a). Suppose $f(0) < 0$ (equivalently (11) holds). Under this condition, $\mathcal{E}^{ex}$ has an unique critical value in $\mathcal{S}$, which we denote by α_*. It remains to verify whether α_* is the minimum of $\mathcal{E}^{ex}$. From (9), we have

$$\frac{\partial^2 \mathcal{E}^{ex}}{\partial \alpha^2} = \frac{4h(\alpha)}{g^3(\alpha)} f(\alpha) + \frac{1}{g^2(\alpha)} f'(\alpha). \tag{12}$$

Hence using the fact that $f(\alpha_*) = 0$, and also $f'(\alpha) > 0$ for $\alpha \in \mathcal{S}$, we get

$$\frac{\partial^2 \mathcal{E}^{ex}(\alpha_*)}{\partial \alpha^2} = \frac{1}{g^2(\alpha_*)} f'(\alpha_*) > 0.$$

Hence, $\mathcal{E}^{ex}$ reaches its infimum over $\mathcal{S}$ at $\alpha_* > 0$ (and does not approach it as $\alpha \to 0$), and so

$$\inf_{\alpha \in \mathcal{S}} \mathcal{E}^{ex}(\alpha, \cdots) < \mathcal{E}^{ex}(0_+, \cdots) = \sum_{i=1}^{N} \frac{\rho_i \lambda_i}{2a_i - a_i^2}.$$

Thus when (11) holds, LMS can track the system according to the criterion in (5).

b). Suppose $f(0) \geq 0$ (equivalently (11) fails). Then $f(\alpha) > 0$ for $\alpha \in \mathcal{S}$. Since $g(\alpha) > 0$ for $\alpha \in \mathcal{S}$, we have from (9) that $\partial \mathcal{E}^{ex} / \partial \alpha > 0$ on $\mathcal{S}$, so that $\mathcal{E}^{ex}$ is monotonically strictly increasing on $\mathcal{S}$. Hence $\mathcal{E}^{ex}$ approaches its infimum over $\mathcal{S}$ as $\alpha \to 0$, and so

$$\inf_{\alpha \in \mathcal{S}} \mathcal{E}^{ex}(\alpha, \cdots) = \mathcal{E}^{ex}(0_+, \cdots) = \sum_{i=1}^{N} \frac{\rho_i \lambda_i}{2a_i - a_i^2}.$$

Thus when (11) fails, LMS can't track the system according to our criterion in (5).

Combining a) and b) proves the theorem. ∎

REMARK: The proof of Theorem 1 actually shows that when the AR parameter settings are trackable, there is an optimum (stable) step-size, and when the settings are untrackable, there is no optimum (stable) step-size and the MSE is minimized only in the limit as the step-size approaches zero. The numerical examples of Section 4 will demonstrate that the trackable region is nontrivial, in that certain AR parameter settings do in fact lead to MSE of LMS which is strictly larger than the MSE of the simple prior estimate ($\hat{y}_k = 0$), for all stable step-size (these parameter settings correspond to fast enough system variations, i.e., large enough values for the $a_i' s$).

Theorem 1 provides a necessary and sufficient condition for trackability. The following corollaries give some additional insight into the structure of the trackable region. Due to limited space, the proofs are omitted.

Corollary 1 *$\Re$ satisfies $\Re \supset (0, a_{1,*}) \times \cdots \times (0, a_{N,*})$, where $a_{i,*}$ is the unique solution of*

$$a_i^3 - 2a_i^2 - \frac{(\operatorname{tr} \mathbf{R}_x + 4\lambda_i)\, \rho_i}{\sigma_n^2} a_i + \frac{2\lambda_i \rho_i}{\sigma_n^2} = 0, \tag{13}$$

with $a_i \in (0,1)$, for $i = 1, \ldots, N$.

Corollary 2 *Let $(a_1, a_2, \cdots, a_N)$, $(\tilde{a}_1, \tilde{a}_2, \cdots, \tilde{a}_N) \in (0,1)^N$ be such that $\tilde{a}_i \leq a_i$ for all $i = 1, 2, \cdots, N$. If $(a_1, a_2, \cdots, a_N)$ is within the trackable region $\Re$, so is $(\tilde{a}_1, \tilde{a}_2, \cdots, \tilde{a}_N)$.*

Corollary 3 *Let $(a_1, a_2, \cdots, a_N)$, $(\tilde{a}_1, \tilde{a}_2, \cdots, \tilde{a}_N) \in (0,1)^N$ be such that $\tilde{a}_i \geq a_i$ for all $i = 1, 2, \cdots, N$. If $(a_1, a_2, \cdots, a_N)$ isn't in the trackable region $\Re$, then neither is $(\tilde{a}_1, \tilde{a}_2, \cdots, \tilde{a}_N)$.*

Corollary 4 *Let $\Re(\sigma_n^2)$ denote the dependence of the trackable region on the noise variance σ_n^2. Then $\Re(\mu) \supset \Re(\nu)$ whenever $\mu < \nu$.*

REMARK:

1. Corollary 1 shows that a system with slow enough variations (i.e. small enough $a_i' s$) is trackable by LMS.
2. Corollaries 2 and 3 show that if a system is trackable by LMS, then any system with (uniformly) slower variation is also trackable; and if a system is not trackable, then any system with (uniformly) faster variations is not trackable, either.
3. Corollary 4 shows that the trackable region expands as the noise variance decreases. This was unclear directly from Theorem 1, since there are terms in (7) which individually expand or contract the region as the noise variance decreases, apparently depending on the values of the other system parameters. As the noise variance decreases, one could think the system output variations are harder to track because they are less "smoothed" by the noise. However, one could also argue that system output variations should be easier to track, because they are less "hidden" by the noise (we believe this makes more sense). Corollary 4 shows the latter effect dominates.

4 Simulation Results

We present some simulation results to further demonstrate and compare with the analytical results derived in Sections 3. The simplest case is when $N = 1$. Denote $a = a_1$, $\sigma_x^2 = \lambda_1$ and $\rho = \rho_1$. Based on Theorem 1, an argument similar to the proof of Corollary 1 shows that $\Re = (0, a_*)$, where a_* is the unique solution of

$$a^3 - 2a^2 - \frac{5\sigma_x^2\rho}{\sigma_n^2}a + \frac{2\sigma_x^2\rho}{\sigma_n^2} = 0,$$

with $a \in (0,1)$. Note that this result is different from Corollary 1, where only inclusion holds. In Fig. 2 we show a_* as a function of $\sigma_x^2\rho/\sigma_n^2$. Observe that the trackable region expands as the noise variance decreases, as expected (Corollary 4).

It is interesting to determine the optimal step-size and the corresponding minimal excess MSE for $a \in \Re$. Now $\mathcal{E}^{ex}$ is minimized over $\alpha \in \mathcal{S}$ at its unique critical value α_* in $\mathcal{S}$, where $\mathcal{S} = (0, 2/3\sigma_x^2)$. Using (9) the optimal step-size is

$$\alpha_* = \frac{\sqrt{9(\frac{\sigma_x^2\rho}{\sigma_n^2})^2 + (4-a^2)\frac{\sigma_x^2\rho}{\sigma_n^2}} - a(2-a) - 3\frac{\sigma_x^2\rho}{\sigma_n^2}}{(1-a)(2-a)\sigma_x^2}.$$

From here, we can plug α_* into (8) to obtain the minimal excess MSE.

Now, we take $\sigma_x^2 = 1, \rho = 0.02, \sigma_n^2 = 0.01$, so that the maximum trackable rate is $a_* = 0.3769$. If $a = 0.01$, the system should be trackable, according to Theorem 1. The simulated MSE curve based on averaging over 100 independent runs is shown in Fig. 3. The computed MSE curve using (8) is also depicted for comparison, and is very close to the simulation results. In this example, $\mathrm{Var}\,[y_k] = 1.01$ is significantly greater than minimum MSE ≈ 0.072, so the system is therefore within the trackable region according to Definition 1. Still for $N = 1$, we take $\sigma_x^2 = 1, \rho = 0.2, \sigma_n^2 = 0.1$, whence the maximum trackable rate is again $a_* = 0.3769$. If $a = 0.4$ the system should not be trackable. The MSE curves are shown in Fig. 4, from which we see that the minimum MSE is achieved as the step-size tends to zero. As per the Remark following Theorem 1, this corresponds to the system outside the trackable region.

Next we consider some numerical examples with $N = 2$. When $\sigma_n^2 = 1$, $\mathbf{R}_x = \mathrm{diag}\{3, 2\}$ and $\mathbf{D} = \mathrm{diag}\{0.3, 0.1\}$, the trackable region from Theorem 1 is calculated numerically and shown in Fig. 5, where the darker area indicates the subset of the trackable region determined by the sufficient condition in Corollary 1 (here $a_{1,*} = 0.3193$ and $a_{2,*} = 0.2336$). It is easy to see that the sufficient condition is quite conservative. Now, set $(a_1, a_2) = (0.20, 0.15)$, the system should be trackable. From the MSE curves shown in Fig. 7, the minimum MSE is about 3.89, which is smaller than $\mathrm{Var}\,[y_k] = 4.22$, confirming that the system is trackable. We also test $(a_1, a_2) = (0.80, 0.10)$ or $(a_1, a_2) = (0.25, 0.80)$, where one of the taps varies extremely fast, but the system should still be trackable, which is indeed verified by simulation results. If $(a_1, a_2) = (0.50, 0.50)$, the system should not be trackable. The MSE curves shown in Fig. 8 also confirm this.

The trackable region when $\mathbf{R}_x = \mathrm{diag}\{8, 2\}$, $\mathbf{D} = \mathrm{diag}\{10, 0.1\}$ and σ_n^2 unchanged is shown in Fig. 6. In this case, $a_{1,*} = 0.3840$ and $a_{2,*} = 0.1727$.

We remark that the trackable region $\Re$ for $N \geq 2$ looks very different from $N = 1$. In fact, when $N = 2$, if one tap varies slowly enough, the other tap may have a very fast variation and the system can still be tracked.

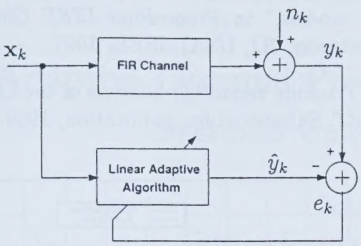


Figure 1: System Model.

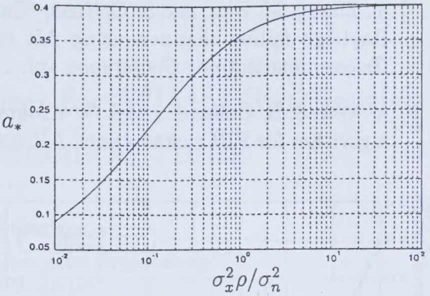


Figure 2: Maximum trackable rate a_* as a function of $\sigma_x^2\rho/\sigma_n^2$.

5 Conclusions

In this paper we examined several issues in measuring and classifying the tracking performance of an adaptive algorithm in identifying a time-varying system. A new criterion is proposed and then used to define the trackable region where the LMS algorithm can track an FIR system with AR coefficients. An explicit description of the trackable region is derived based on generalized results which characterize the mean-square stability and excess mean square error of LMS for this case. The specification of the trackable region involves the optimization of the excess MSE subject to a stability constraint. The qualitative character of the trackability region is explored via analysis and simulations.

Throughout this article, no assumptions on "small step-size" or "slow varying system" are made. In fact, this work deals with both slow and fast system variations. Instead of studying the effect of step-size on the tracking performance of the LMS algorithm, we investigate the tracking capability of the LMS algorithm based on its optimal performance. The point is to understand how and when performance breaks down as one moves to larger step-size and/or faster varying systems. The technique can potentially be applied to examine the trackable regions of other time-varying system models and adaptive algorithms.

References

[1] S. Haykin, *Adaptive Filter Theory.* Englewood Cliffs, NJ: Prentice–Hall, 1991.

[2] L. Guo, L. Ljung, and G. Wang, "Necessary and sufficient conditions for stability of LMS," *IEEE Trans. Automatic Control*, vol. 42, pp. 761–770, June 1997.

[3] V. Solo, "The error variance of LMS with time-varying weights," *IEEE Trans. Signal Processing*, vol. 40, pp. 803–813, April 1992.

[4] O. Macchi, *Adaptive processing : The Least Mean Square approach with applications in transmission.* New York: John Wiley and Sons, 1995.

[5] V. Solo and X. Kong, *Adaptive Signal Processing Algorithms: Stability and Performance.* Englewood Cliffs, NJ: Prentice Hall, 1995.

[6] H. Meyr, M. Moeneclaey, and S. A. Fechtel, *Digital Communication Receivers: Synchronization, Channel Estimation, and Signal processing.* John Wiley & Sons, 1998.

[7] Y. Zhang, M. Fitz, and S. Gelfand, "Soft output demodulation on frequency-selective Rayleigh fading channels using AR channel models," in *Proceedings IEEE Global Telecommunications Conference*, vol. 1, (Piscataway, NJ, USA), IEEE, 1997.

[8] Y. Wei, S. B. Gelfand, and J. V. Krogmeier, "Tracking capability analysis of the LMS algorithm for FIR systems with AR coefficient." Submitted for publication, 1999.

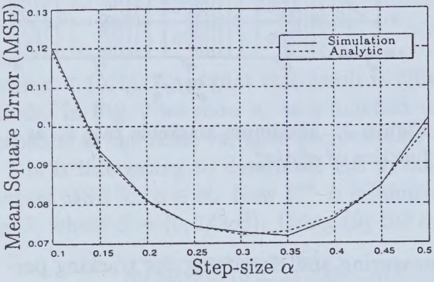


Figure 3: MSE curves when $N = 1, \sigma_x^2 = 1, \rho = 0.02, \sigma_n^2 = 0.01, a = 0.01$.

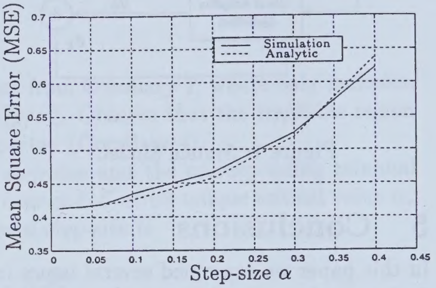


Figure 4: MSE curves when $N = 1, \sigma_x^2 = 1, \rho = 0.2, \sigma_n^2 = 0.1, a = 0.4$.

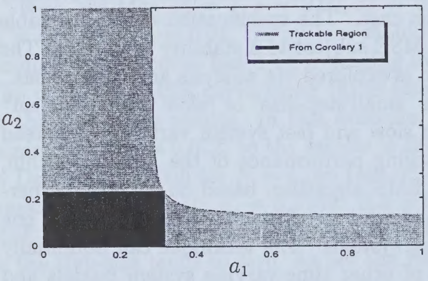


Figure 5: Trackable region for $N = 2, \sigma_n^2 = 1, \mathbf{R}_x = \text{diag}\{3, 2\}$ and $\mathbf{D} = \text{diag}\{0.3, 0.1\}$.

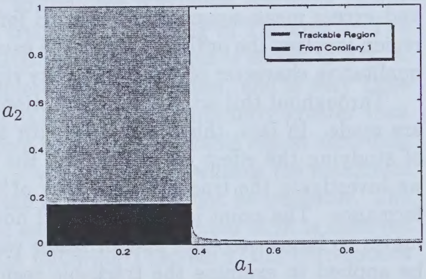


Figure 6: Trackable region for $N = 2, \sigma_n^2 = 1, \mathbf{R}_x = \text{diag}\{8, 2\}$ and $\mathbf{D} = \text{diag}\{10, 0.1\}$.

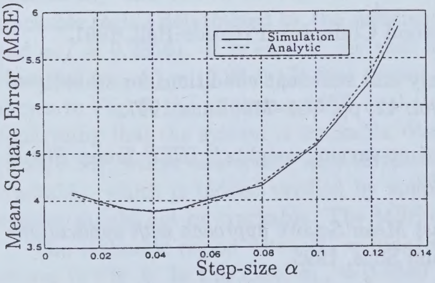


Figure 7: MSE curves when $N = 2, \sigma_n^2 = 1, \mathbf{R}_x = \text{diag}\{3, 2\}, \mathbf{D} = \text{diag}\{0.3, 0.1\}$ and $(a_1, a_2) = (0.20, 0.15)$.

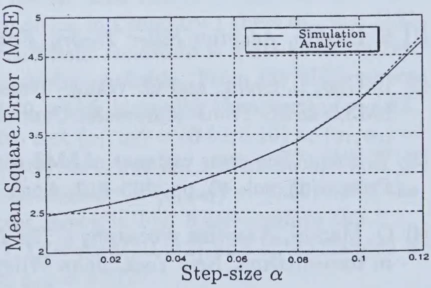


Figure 8: MSE curves when $N = 2, \sigma_n^2 = 1, \mathbf{R}_x = \text{diag}\{3, 2\}, \mathbf{D} = \text{diag}\{0.3, 0.1\}$ and $(a_1, a_2) = (0.50, 0.50)$.

Adaptive Optimal Prediction for MIMO Stochastic Systems Using Canonical Forms*

Behnam Shahrrava
Dept. of Electrical and Computer Engineering
University of Waterloo
Waterloo, Ontario, Canada, N2L 3G1
shahrva@kingcong.uwaterloo.ca

J. D. Aplevich
Dept. of Electrical and Computer Engineering
University of Waterloo
Waterloo, Ontario, Canada, N2L 3G1
aplevich@kingcong.uwaterloo.ca

Abstract

An adaptive predictor for decouplable systems which is optimal at each sample time is proposed, based on priori knowledge of the structure of the interactor matrix and a canonical form. The adaptive predictor includes the uncertainty associated with the parameter and state estimates. Moreover, it is shown that no optimal solution to indirect adaptive prediction exists for nondecouplable systems having unknown interactor matrix.

1 Introduction

Adaptive prediction in the context of multi-input, multi-output (MIMO) systems is useful in applications such as predictive control of processes [3], prediction of maneuvering target trajectories [4], river flow prediction [9], and electric power system load forecasting [7]. Most of the existing adaptive prediction methods use the certainty equivalence principle. For systems having diagonal or nondiagonal interactor matrix, conventional approaches solve the adaptive optimal prediction problem approximately. However, for systems having diagonal interactor matrix the adaptive predictor in this paper is optimal at each sample time. The approach uses knowledge of the structure of the interactor matrix and a canonical form. It will also be shown that no optimal solution to indirect adaptive prediction exists for nondecouplable systems having unknown interactor matrix. The results obtained in this paper for MIMO systems are an extension of for single-input, single-output (SISO) systems.

The paper is organized as follows. In Section 2 the system model is described. The interactor matrix, which is a generalization of the relative degree of a scalar system, is also reviewed. Section 3 discusses the optimality criterion of interest. In Section 4,

---

*This research is supported in part by Natural Sciences and Engineering Research Council of Canada Grant number 7380.

using state variables it is shown how the system model can be transformed into a special model similar to one proposed by Salut [10, 2]. Section 5 applies a Kalman filter to the canonical model to provide optimal joint parameter and state estimation. Then in Section 6 it is shown how the system model can be transformed into several predictor forms. In Section 7 the entries of the interactor matrix are related to the canonical ARX model. Section 8 contains the main results. An adaptive optimal predictor for systems having diagonal interactor matrix is constructed, as for the SISO case in [12].

2 System description

A linear time-invariant MIMO system is assumed to be represented by the following canonical ARX model:

$$A(q^{-1})y_k = B(q^{-1})u_k + e_k \tag{1}$$

where y_k is $p \times 1$, u_k is $m \times 1$, and e_k is $p \times 1$. $A(q^{-1})$, $B(q^{-1})$ are $p \times p$ and $p \times m$ polynomial matrices in the unit delay operator q^{-1} with the following forms:

$$\begin{aligned} A(q^{-1}) &= A_0 + A_1 q^{-1} + \cdots + A_v q^{-v}, && \text{(2a)} \\ B(q^{-1}) &= B_0 + B_1 q^{-1} + \cdots + B_v q^{-v} && \text{(2b)} \end{aligned}$$

where v is the maximum of the degrees of the polynomials in $A(q^{-1})$ and $B(q^{-1})$. The independent sequence $\{e_k\}$ is a gaussian zero mean white noise process with given covariance

$$E\{e_k e_l^T\} = R_k \delta_{k-l} \ , \quad R_k > 0 \tag{3}$$

where δ_{k-l} is the Kronecker delta.

It is assumed that the input-output transfer matrix $T(z) = A(z^{-1})^{-1}B(z^{-1})$ is full rank and strictly proper. The latter assumption ensures that there is a delay of at least one unit between each input and each output. Hence, for a strictly proper system B_0 is zero in (2b). The matrix $\xi(q)$, known as the interactor matrix [14], is a $p \times p$ nonsingular Hurwitz lower left triangular matrix, of the form

$$\xi(z) = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ h_{21}(z) & 1 & \cdots & 0 \\ \vdots & \vdots & & \vdots \\ h_{p1}(z) & h_{p2}(z) & \cdots & 1 \end{bmatrix} \mathrm{diag}[z^{d_1}, z^{d_2}, \ldots, z^{d_p}] \tag{4}$$

where $h_{ij}(z)$ is divisible by z (or is zero); and the $d_i \neq 0$ are positive integers satisfying

$$\lim_{z \to \infty} \xi(z)T(z) = K_T \tag{5}$$

where K_T is a full rank constant matrix which can be seen as the analog of the high-frequency gain in the SISO case.

The interactor matrix defines the infinite structure of the the transfer matrix and in particular the infinite zeros which reduce to the delay in the SISO case. Therefore, the interactor matrix is part of the transfer matrix that cannot be modified by state feedback. For instance, it can be shown that the system is decouplable iff the interactor matrix is diagonal [13].

3 Minimum mean square error prediction

The aim is to obtain information about the filtered output $\xi(q)y_k$ at time k. In obtaining the information, measurements up to and including time k, denoted by $\mathcal{F}_k$, can be used. The criterion of interest is the minimum mean square error prediction based on the minimization of the following cost function:

$$J = E\{\|\xi(q)y_k - E\{\xi(q)y_k | \mathcal{F}_k\}\|^2 | \mathcal{F}_k\} \tag{6}$$

where $E\{\xi(q)y_k | \mathcal{F}_k\}$ is the optimal prediction of the filtered output $\xi(q)y_k$ at time k in a minimum variance sense. Hence the objective is to produce $E\{\xi(q)y_k | \mathcal{F}_k\}$.

In the following sections it will be shown how this optimal problem can be solved without using approximations.

4 Minimal canonical representations

A brief introduction to the class of minimal canonical representations required is given in this section, discussed in detail by Aplevich [1, 2].

First, by introducing auxiliary variables in a vector x_k, the canonical ARX model (1) is written as

$$\begin{aligned} x_{k+1} &= E_* x_k - A_* y_k + B_* u_k && \text{(7a)} \\ 0 &= E_0 x_k - A_0 y_k + B_0 u_k + e_k && \text{(7b)} \end{aligned}$$

where $E_*, E_0, A_*, A_0, B_*, B_0$ are matrices with the following properties:

- For causal discrete-time systems, the row degrees of q^{-1} in $A(q^{-1})$ are no less than the corresponding row degrees of $B(q^{-1})$, and $A(q^{-1})$ is invertible. Thus A_0 is invertible, and furthermore in the canonical form considered, A_0 is upper right triangular with unit diagonal entries.

- The integers n_i, $i = 1, \ldots, p$ are the *observability indices* of the system and $n = n_1 + n_2 + \cdots + n_p$.

- The matrix $E = [E_*^T, E_0^T]^T$ is zero except for a positive unit element in each column. The unit elements, called pivots, in the n non-zero rows are to the right of pivot elements in superior rows. The locations of the pivots are uniquely defined by n_i such that if row i of E_0 is denoted by E_{0i}, then $E_{0i}E_*^{n_i}$ is the lowest zero row in the matrix $\tilde{E} = [(E_0 E_*^n)^T, \ldots, (E_0 E_*)^T, E_0^T]^T$. Thus E contains zero rows $E_{0i}E_*^{n_i}$ together with the n non-zero rows of $\tilde{E}$, so that each row of E is of the form $E_{0i}E_*^j$, and may be referred to as row (i, j).

- The non-identically-zero, non-unit entries in $A = [A_*^T, A_0^T]^T$ are designated $a_{ik}^{(j)}$ and their locations are uniquely determined by $\{n_i\}$. Each $a_{ik}^{(j)}$ is in row (i, j) of A, using the row indexing of E above, and column $k = 1, 2, \ldots, p$. There is an entry $a_{ik}^{(j)}$ for each non-zero row of the form $E_{0k}E_*^{n_i - j}$ below $E_{0i}E_*^{n_i}$ in $\tilde{E}$, that is, for each non-zero row with index $(k, n_i - j)$ below row (i, n_i) in E.

- For causal systems the matrix $B = [B_*^T, B_0^T]^T$ does not have a special structure and the subscripts of its entries $b_{ik}^{(j)}$ are as for $a_{ik}^{(j)}$ except that $k = 1, 2, \ldots, m$. For a strictly proper system, B_0 is zero.

It has been shown by Aplevich [1] that the parameters $\{a_{ik}^{(j)}\}$ and $\{b_{ik}^{(j)}\}$ in (7) are identical to those in a canonical ARX model (1). Specifically, the parameters $a_{ik}^{(j)}$ and $b_{ik}^{(j)}$ in (7) are the ik entries of certain canonical coefficient matrices A_j and B_j in equations (2a)-(2b), respectively.

It has been proved by Aplevich [1] that with $[-A_0\ B_0]$ in reduced row echelon form, the dynamical system above, (7), is minimal iff it possesses the two properties:

Property 1. $[E_* - \lambda I \quad -A_* \ \ B_*]$ has full row rank for all complex λ.

Property 2. $[E_*^T - \lambda I, E_0^T]^T$ has full column rank for all complex λ.

Note that the model (7) is not a standard state-space model; however, a state-space model can easily be obtained by substituting (7b) into (7a) as follows:

$$\begin{aligned} x_{k+1} &= \bar{A}x_k + \bar{B}u_k + w_k && (8a) \\ y_k &= \bar{C}x_k + \bar{D}u_k + e_k && (8b) \end{aligned}$$

where

$$\bar{A} = E_* - A_* A_0^{-1} E_0\,, \quad \bar{B} = B_* - A_* A_0^{-1} B_0\,, \quad \bar{C} = A_0^{-1} E_0\,, \quad \bar{D} = A_0^{-1} B_0\,, \qquad (8c)$$

and

$$w_k = -A_* e_k\,, \quad E\left\{ \begin{bmatrix} w_k \\ e_k \end{bmatrix} \begin{bmatrix} w_l \\ e_l \end{bmatrix}^T \right\} = \begin{bmatrix} Q_k & S_k \\ S_k^T & R_k \end{bmatrix} \delta_{k-l}\,. \qquad (8d)$$

In the next section, it will become clear why model (7) is used to determine the optimal joint parameter and state of the system, rather than using the standard state-space model (8).

The following example further illustrates the relations among the canonical model (7), canonical ARX model (1), and state-space model (8).

Example 4.1: Consider a 2-input, 2-output strictly proper system with $\{n_i\} = \{2, 2\}$. The structure of the canonical form for this system, obtained by applying the algorithms developed in [1], is given as in (7) with

$$\begin{bmatrix} E_* & -A_* & B_* \\ E_0 & -A_0 & B_0 \end{bmatrix} = \left[\begin{array}{cccccc|cc} 0 & 0 & 0 & 0 & -a_{11}^{(2)} & -a_{12}^{(2)} & b_{11}^{(2)} & b_{12}^{(2)} \\ 0 & 0 & 0 & 0 & -a_{21}^{(2)} & -a_{22}^{(2)} & b_{21}^{(2)} & b_{22}^{(2)} \\ 1 & 0 & 0 & 0 & -a_{11}^{(1)} & 0 & b_{11}^{(1)} & b_{12}^{(1)} \\ 0 & 1 & 0 & 0 & 0 & -a_{22}^{(1)} & b_{21}^{(1)} & b_{22}^{(1)} \\ \hline 0 & 0 & 1 & 0 & -1 & -a_{12}^{(0)} & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & -1 & 0 & 0 \end{array} \right]. \qquad (9)$$

The corresponding canonical ARX model (1) can be obtained from (9) by inspection, where

$$\begin{aligned} A(q^{-1}) &= \begin{bmatrix} 1 & a_{12}^{(0)} \\ 0 & 1 \end{bmatrix} + \begin{bmatrix} a_{11}^{(1)} & 0 \\ 0 & a_{22}^{(1)} \end{bmatrix} q^{-1} + \begin{bmatrix} a_{11}^{(2)} & a_{12}^{(2)} \\ a_{21}^{(2)} & a_{22}^{(2)} \end{bmatrix} q^{-2} \\ B(q^{-1}) &= \begin{bmatrix} b_{11}^{(1)} & b_{12}^{(1)} \\ b_{21}^{(1)} & b_{22}^{(1)} \end{bmatrix} q^{-1} + \begin{bmatrix} b_{11}^{(2)} & b_{12}^{(2)} \\ b_{21}^{(2)} & b_{22}^{(2)} \end{bmatrix} q^{-2}\,. \end{aligned}$$

From (8c), the system matrices $\{\bar{A}, \bar{B}, \bar{C}, \bar{D}\}$ in the corresponding state-space model are:

$$\bar{A} = \begin{bmatrix} 0 & 0 & -a_{11}^{(2)} & a_{11}^{(2)}a_{12}^{(0)} - a_{12}^{(2)} \\ 0 & 0 & -a_{21}^{(2)} & a_{21}^{(2)}a_{12}^{(0)} - a_{22}^{(2)} \\ 1 & 0 & -a_{11}^{(1)} & a_{11}^{(2)}a_{12}^{(0)} \\ 0 & 1 & 0 & -a_{22}^{(1)} \end{bmatrix}, \qquad \bar{B} = \begin{bmatrix} b_{11}^{(2)} & b_{12}^{(2)} \\ b_{12}^{(2)} & b_{22}^{(2)} \\ b_{11}^{(1)} & b_{12}^{(1)} \\ b_{21}^{(1)} & b_{22}^{(1)} \end{bmatrix}, \tag{10}$$
$$\bar{C} = \begin{bmatrix} 0 & 0 & 1 & -a_{12}^{(0)} \\ 0 & 0 & 0 & 1 \end{bmatrix}, \qquad \bar{D} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}.$$

5 Parameter and state estimation

In this section, the optimal joint parameter and state estimate of the system is described [10, 11]. For $B_0 = 0$, the model (7) can be written as

$$x_{k+1} = E_* x_k - A_* y_k + B_* u_k \tag{11a}$$
$$y_k = E_0 x_k + (I - A_0) y_k + e_k \tag{11b}$$

where A_0 is unit triangular. Let θ contain the unknown parameters of the system (7). Then the model (11) can be rearranged in the form

$$x_{k+1} = E_* x_k + G_* \theta \tag{12a}$$
$$y_k = E_0 x_k + G_0 \theta + e_k \tag{12b}$$

where the matrices G_* and G_0 are constructed according to the order of the parameters in θ. Since for a time-invariant system $\theta_{k+1} = \theta_k = \theta$, by appending the parameter vector θ_k to the state vector x_k and using the model (12), the following dynamical model can be used for the estimation process:

$$s_{k+1} = \Phi_k s_k \tag{13a}$$
$$y_k = H_k s_k + e_k \tag{13b}$$

where

$$s_k = \begin{bmatrix} x_k \\ \theta_k \end{bmatrix}, \quad \Phi_k = \begin{bmatrix} E_* & G_* \\ 0 & I \end{bmatrix}, \quad H_k = \begin{bmatrix} E_0 & G_0 \end{bmatrix}. \tag{14}$$

The importance of the representation (13) above is that all unknowns appear linearly when an input-output record is available. Now, in order to apply a Kalman filter to the model (13), it is assumed that s_0 and $\{e_k\}$ are independent and gaussian; s_0 is $\mathcal{N}(\bar{s}_0, \Sigma_0)$; $\{e_k\}$ is zero mean, with covariance $R_k \delta_{k-l}$. Under these assumptions, the conditional distribution of s_{k+1} given $\mathcal{F}_k$ is gaussian with mean $\hat{s}_{k+1|k}$ and covariance $P_{k+1|k}$, see [10],

$$\hat{s}_{k+1|k} = E\{s_{k+1} \,|\, \mathcal{F}_k\}, \tag{15}$$
$$P_{k+1|k} = E\{(s_{k+1} - \hat{s}_{k+1|k})(s_{k+1} - \hat{s}_{k+1|k})^T \,|\, \mathcal{F}_k\}. \tag{16}$$

Then $\hat{s}_{k+1|k}$ and $P_{k+1|k}$ can be obtained using a Kalman filter as follows:

$$\hat{s}_{k+1|k} = \Phi_k \hat{s}_{k|k-1} + K_k (y_k - H_k \hat{s}_{k|k-1}), \tag{17a}$$
$$K_k = \Phi_k P_{k|k-1} H_k^T (H_k P_{k|k-1} H_k^T + R_k)^{-1}, \tag{17b}$$
$$P_{k+1|k} = \Phi_k P_{k|k-1} \Phi_k^T - K_k (H_k P_{k|k-1} H_k^T + R_k) K_k^T, \tag{17c}$$

with initial conditions $\hat{s}_{0|-1} = \bar{s}_0$ and $P_{0|-1} = \Sigma_0 \geq 0$.

6 MIMO predictor forms

Generalizing [12], the model (7) will be expressed in a predictor form. Then it will be shown that the conventional predictor (Goodwin and Sin [5]) is a special case of the predictor derived here.

The following lemma shows how a predictor form for a canonical MIMO model can be derived and how the filtered output $\xi(q)y_k$ can be determined.

Lemma 1 *For a strictly proper system having the canonical model (7), the filtered output* $\xi(q)y_k$ *can be expressed in the following predictor form:*

$$\xi(q)y_k = \sum_{l=0}^{d'-1} \Lambda_l \bar{C}\bar{A}^{d'-l-1} x_{k+1} + \sum_{i=0}^{d'-1} F_i e_{k+d'-i} \tag{18}$$

where

$$\begin{aligned} x_{k+1} &= E_*^{k+1} x_0 - \sum_{i=0}^{k} E_*^{k-i} A_* y_i + \sum_{i=0}^{k} E_*^{k-i} B_* u_i && (19) \\ \xi(q) &= \Lambda_0 q^{d'} + \Lambda_1 q^{d'-1} + \cdots + \Lambda_{d'-1} q && (20) \\ d' &= \textit{maximum advance in } \xi(q) && (21) \end{aligned}$$

for $k \geq 0$ *and* $d' \geq 1$. *Furthermore, the coefficient matrices* $F_0, \ldots, F_{d-1}$ *can be computed as follows:*

$$F_i = \begin{cases} \Lambda_0 A_0^{-1}, & i = 0 \\ (\Lambda_i - \sum_{l=0}^{i-1} \Lambda_l \bar{C}\bar{A}^{i-l-1} A_*) A_0^{-1}, & i = 1, \ldots, d' - 1. \end{cases} \tag{22}$$

Proof: From (7) and using a sequential method [11], we obtain

$$\xi(q)y_k = \sum_{l=0}^{d'-1} \Lambda_l \bar{C}\bar{A}^{d'-l-1} x_{k+1} + \sum_{i=k+1}^{k+d'-1} \beta_{k-i} u_i + \sum_{i=0}^{d'-1} F_i e_{k+d'-i} \tag{23}$$

for $k \geq 0$, $d' \geq 1$. Now substituting (19) into (23) gives

$$\xi(q)y_k = L_k x_0 + \sum_{i=0}^{k} \alpha_{k-i} y_i + \sum_{i=0}^{k+d'-1} \beta_{k-i} u_i + \sum_{i=0}^{d'-1} F_i e_{k+d'-i} \tag{24}$$

where

$$\begin{aligned} L_k &= \textstyle\sum_{l=0}^{d'-1} \Lambda_l \bar{C}\bar{A}^{d'-l-1} E_*^{k+1} \\ \alpha_{k-i} &= -\textstyle\sum_{l=0}^{d'-1} \Lambda_l \bar{C}\bar{A}^{d'-l-1} E_*^{k-i} A_*, \qquad i = 0, \ldots, k \\ \beta_{k-i} &= \begin{cases} \sum_{l=0}^{d'-1} \Lambda_l \bar{C}\bar{A}^{d'-l-1} E_*^{k-i} B_* & i = 0, \ldots, k \\ \sum_{l=0}^{k+d'-1-i} \Lambda_l \bar{C}\bar{A}^{k+d'-l-i-1} B_* & i = k+1, \ldots, k+d'-1 \end{cases} \\ F_i &= \begin{cases} \Lambda_0 A_0^{-1}, & i = 0 \\ (\Lambda_i - \sum_{l=0}^{i-1} \Lambda_l \bar{C}\bar{A}^{i-l-1} A_*) A_0^{-1}, & i = 1, \ldots, d' - 1 \end{cases} \end{aligned} \tag{25}$$

for $k \geq 0$ and $d' \geq 1$.

Now the coefficient matrices $\{\Lambda_0, \ldots, \Lambda_{d'-1}\}$ of $\xi(q)$ can be chosen so that the coefficients of $u_{k+1}, \cdots, u_{k+d'-1}$ are zero and also the coefficient of u_k is nonsingular (in the SISO case the predictor coefficient of u_k is nonzero),

$$\begin{aligned} \beta_{-d'+1} &= \cdots = \beta_{-1} = 0 && (26a) \\ \beta_0 &= \textit{full rank} && (26b) \end{aligned}$$

where from (25)

$$\begin{aligned} \beta_i &= \sum_{l=0}^{d'-1+i} \Lambda_l \bar{C} \bar{A}^{d'-l+i-1} B_* \,, \qquad i = -d'+1, \ldots, -1 && (26c) \\ \beta_0 &= \sum_{l=0}^{d'-1} \Lambda_l \bar{C} \bar{A}^{d'-l-1} B_* \,. && (26d) \end{aligned}$$

After substituting (26) into (23), equation (18) follows immediately.

In the following lemma the predictor proposed in Lemma 1 will be expressed in an alternative form that will be used later.

Lemma 2 *The predictor model (18) in Lemma 1 can be written into the following equivalent form together with the initial condition x_0:*

$$\xi(q) y_k = L_k x_0 + \sum_{i=0}^{k} \alpha_{k-i} y_i + \sum_{i=0}^{k} \beta_{k-i} u_i + \sum_{i=0}^{d'-1} F_i e_{k+d'-i} \tag{27}$$

for $k \geq 0$ and $d' \geq 1$.

Proof: Follows by substituting (26a) into (24).

Note that as expected the filtered output $\xi(q)y_k$ is a function of the system outputs and the system inputs up to and including time k as well as the future (unpredictable) noise $\sum_{i=0}^{d'-1} F_i e_{k+d'-i}$.

Now it will be shown how the predictor model (27) can be related to the conventional predictor [5].

Lemma 3 *The predictor model (18) can be expressed in the following form*

$$\xi(q) y_k = \alpha(q^{-1}) y_k + \beta(q^{-1}) u_k + F(q) e_k \tag{28a}$$

where

$$\begin{aligned} \alpha(q^{-1}) &= \alpha_0 + \alpha_1 q^{-1} + \cdots + \alpha_{v-1} q^{-(v-1)} && (28b) \\ \beta(q^{-1}) &= \beta_0 + \beta_1 q^{-1} + \cdots + \beta_{v-1} q^{-(v-1)} && (28c) \\ F(q) &= F_0 q^{d'} + \cdots + F_{d'-1} q && (28d) \end{aligned}$$

and v in the ARX model (1) is the maximum of the degrees of the polynomials in $A(q^{-1})$ and $B(q^{-1})$, or equivalently $v = \max_{i=1,\ldots,p}\{n_i\}$ in the canonical model (7).

Proof: Because of the special structure of matrix E_*

$$E_*^l = 0_{n \times n} \qquad \text{for } l \geq v = \max_{i=1,\ldots,p} \{n_i\} \,, \tag{29}$$

see [2], where the integers n_i, $i = 1, \ldots, p$ are the observability indices of the system, as explained in Section 4. Then for $k \geq v - 1$ the coefficients L_k, $\{\alpha_k, \ldots, \alpha_{k+(v-2)}\}$, and $\{\beta_k, \ldots, \beta_{k+(v-2)}\}$ are zero. After setting these coefficients to zero in (27), equation (28a) follows immediately. Thus the predictor model (28a) is a special case of (27). For zero initial state there is no difference between two models. However, the proposed predictor is capable of dealing with an unknown nonzero initial state in an adaptive case whereas the conventional predictor model lacks such a capability.

7 Determination of the parameters of $\xi(q)$

Using the structure of $\xi(q)$, the structure of the matrices Λ_i in (20) can be specified, and then the unknown entries of the matrices Λ_i are determined by solving the matrix equations (26a). The following example illustrates the procedure.

Example 4.1 (Continued): Assume that the interactor matrix of Example 4.1 has the following structure:

$$\xi(q) = \begin{bmatrix} q & 0 \\ \Lambda_{21}^{(0)} q^2 & q^2 \end{bmatrix} = \Lambda_0 q^2 + \Lambda_1 q = \begin{bmatrix} 0 & 0 \\ \Lambda_{21}^{(0)} & 1 \end{bmatrix} q^2 + \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} q \tag{30}$$

From (26) and using (9), and (10),

$$\Lambda_0 \bar{C} B_* = \begin{bmatrix} 0 & 0 \\ \Lambda_{21}^{(0)} & 1 \end{bmatrix} \begin{bmatrix} 0 & 0 & 1 & -a_{12}^{(0)} \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} b_{11}^{(2)} & b_{12}^{(2)} \\ b_{12}^{(2)} & b_{22}^{(2)} \\ b_{11}^{(1)} & b_{12}^{(1)} \\ b_{21}^{(1)} & b_{22}^{(1)} \end{bmatrix} = 0 \tag{31}$$

where $\Lambda_{21}^{(0)}$ is

$$\Lambda_{21}^{(0)} = \begin{cases} \frac{-b_{21}^{(1)}}{b_{11}^{(1)} - a_{12}^{(0)} b_{21}^{(1)}} & \text{if } b_{11}^{(1)} - a_{12}^{(0)} b_{21}^{(1)} \neq 0 \\ \\ \frac{-b_{22}^{(1)}}{b_{12}^{(1)} - a_{12}^{(0)} b_{22}^{(1)}} & \text{if } b_{12}^{(1)} - a_{12}^{(0)} b_{22}^{(1)} \neq 0 \end{cases} \tag{32}$$

provided that $b_{11}^{(1)} b_{22}^{(1)} - b_{12}^{(1)} b_{21}^{(1)} = 0$; otherwise there is no solution. Note that the interactor matrix of this system would be diagonal if $\Lambda_{12}^{(0)} = 0$, which is equivalent to assuming that $b_{21}^{(1)} = b_{22}^{(1)} = 0$ and $b_{11}^{(1)} \neq 0$ or $b_{12}^{(1)} \neq 0$. The above example shows how the nondiagonal entries of the interactor matrix are nonlinearly related to the parameters of the polynomials of $A(q^{-1})$ and $B(q^{-1})$ in the canonical ARX model (1).

8 General optimal solution

As explained in Section 3 we wish to produce $E\{\xi(q) y_k | \mathcal{F}_k\}$. Taking conditional expectation given $\mathcal{F}_k$ on both sides of (18),

$$E\{\xi(q) y_k | \mathcal{F}_k\} = E\{\sum_{l=0}^{d'-1} \Lambda_l \bar{C} \bar{A}^{d'-l-1} x_{k+1} | \mathcal{F}_k\} \tag{33}$$

since $E\{\sum_{i=0}^{d'-1} F_i e_{k+d'-i} | \mathcal{F}_k\} = 0$.

In Example 4.1, assuming for simplicity that $a_{12}^{(0)} = 0$ and $b_{11}^{(1)} \neq 0$, the nondiagonal entry of the interactor matrix is $\Lambda_{21}^{(0)} = -b_{21}^{(1)}/b_{11}^{(1)}$. Since the expected value of the ratio of two jointly gaussian random variables does not exist, (Ryall and Moore [8]), the conditional expectation in (33) cannot be computed. Therefore, an optimal solution to indirect adaptive prediction for nondecouplable systems does not exist. However, if the system is decouplable, the interactor matrix is diagonal and the computation of $E\{\xi(q)y_k | \mathcal{F}_k\}$ involves the computation of expected values of sums of products of correlated gaussian random variables. The following lemma shows how the expected values of products of correlated gaussian random variables can be determined.

Lemma 4 *If $Z = [z_1, z_2, \ldots, z_n]^T$ is a gaussian random vector with mean vector M and positive-semidefinite covariance matrix Σ, then the higher moments depend only on the elements of the mean vector and the covariance matrix, and can be obtained as follows:*

$$E\{z_1^{r_1} z_2^{r_2} \cdots z_n^{r_n}\} = \left. \frac{\partial^{r_1 + r_2 + \cdots + r_n}}{\partial \lambda_1^{r_1} \partial \lambda_2^{r_2} \cdots \partial \lambda_n^{r_n}} \phi_Z(\lambda) \right|_{\lambda = 0} \tag{34}$$

where $\{r_i\}$ are nonnegative integers, and $\phi_Z(\lambda) = e^{\lambda^T M + \frac{1}{2}\lambda^T \Sigma \lambda}$, which $\phi_Z(\lambda)$ is called the moment generating function for Z.

Proof: See [6].

Using the above lemma, an optimal predictor can be designed for systems having diagonal interactor matrix. For simplicity, the system in Example 4.1 with the following diagonal interactor matrix will be used; extension to a general case is straightforward but lengthy:

$$\xi(q) = \begin{bmatrix} q & 0 \\ 0 & q^2 \end{bmatrix} = \Lambda_0 q^2 + \Lambda_1 q = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} q^2 + \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} q \, . \tag{35}$$

The maximum advance in the interactor matrix above is $d' = 2$, and (33) can be written as follows:

$$E\{\xi(q)y_k | \mathcal{F}_k\} = E\{(\Lambda_0 \bar{C}\bar{A} + \Lambda_1 \bar{C})x_{k+1} | \mathcal{F}_k\} \, , \tag{36}$$

which can be written, using (10) and (35), as

$$E\{\xi(q)y_k | \mathcal{F}_k\} = E\{\begin{bmatrix} 0 & 0 & 1 & -a_{12}^{(0)} \\ 0 & 1 & 0 & -a_{22}^{(1)} \end{bmatrix} x_{k+1} | \mathcal{F}_k\} = E\{\begin{bmatrix} x_{k+1}^{(3)} - a_{12}^{(0)} x_{k+1}^{(4)} \\ x_{k+1}^{(2)} - a_{22}^{(1)} x_{k+1}^{(4)} \end{bmatrix} | \mathcal{F}_k\} \tag{37}$$

where $x_{k+1} = [x_{k+1}^{(1)}, x_{k+1}^{(2)}, x_{k+1}^{(3)}, x_{k+1}^{(4)}]^T$.

Using Lemma 4 the optimal predictor is:

$$E\{\xi(q)y_k | \mathcal{F}_k\} = \begin{bmatrix} \hat{x}_{k+1|k}^{(3)} - \hat{a}_{12,k+1|k}^{(0)} \hat{x}_{k+1|k}^{(4)} - P_{a_{12}^{(0)} x_{k+1}^{(4)} | k} \\ \hat{x}_{k+1|k}^{(2)} - \hat{a}_{22,k+1|k}^{(1)} \hat{x}_{k+1|k}^{(4)} - P_{a_{22}^{(1)} x_{k+1}^{(4)} | k} \end{bmatrix} . \tag{38}$$

As expected, the adaptive optimal predictor (38) includes uncertainty associated with both the parameter and state estimates, as in the SISO case [12].

9 Conclusions

The proposed MIMO adaptive predictor for decouplable systems is optimal at each sample time. The adaptive predictor includes the uncertainty associated with both the parameter and state estimates, whereas the conventional adaptive predictors ignore any uncertainty in the parameter estimates. It was also shown that no optimal solution to indirect adaptive prediction exists for systems having nondiagonal unknown interactor matrix. These results can be considered as the generalization of [12].

References

[1] J. D. Aplevich, "Time-domain Input-Output Representations of Linear Systems," *Automatica*, Vol. 17(No. 3):pp. 509–522, 1981.

[2] J. D. Aplevich, "Implicit linear systems," *Springer-Verlag*, 1991.

[3] D. W. Clarke, P. S. Tuffs, and C. Mohtadi, "Self-tuning control of a difficult process," *Proc. IFAC Symp. on Identification and System Parameter Estimation*, York, pp. 1009-1014, 1985.

[4] G. Favier and A. Smolders, "Adaptive smoother predictors for tracking maneuvering targets," *IEEE Conf. on Decision and Control*, Las Vegas, December, 1984.

[5] G. C. Goodwin and K. S. Sin, "Adaptive filtering prediction and control," *Englewood Cliffs, New Jersey, Prentice-Hall*, 1984.

[6] K. S. Miller, "Multidimensional Gaussian Distributions, " *The SIAM Series in Applied Mathematics, John Wiley and Sons Inc.*, 1964.

[7] M. Nakamura, "Short term load forecasting using daily updated load models," *Automatica*, Vol.21, pp. 729-736, 1985.

[8] T. Ryall and J. B. Moore, "Central Tendency Minimum Variance Adaptive Control," *IEEE Trans. Automat. Contr.*, Vol. 34(No. 3):pp. 367-371, 1989.

[9] Abou-Loukh Sadiq and A. K. Sinha, "Self-tuning predictors with application to river flow prediction," *Int. J. Syst. Sci.*, vol. 17(4), pp. 655-668, 1986.

[10] G. Salut, "Identification optimale des systèmes linéaires stochastiques," *PhD thesis, Université Paul Sabtier, Toulouse, France*, 1976.

[11] B. Shahrrava, "Indirect stochastic adaptive control using optimal joint parameter and state estimation," *Ph.D. thesis, University of Waterloo, Waterloo, Canada*, 1998.

[12] B. Shahrrava and J. D. Aplevich, "An Optimal Solution to Adaptive Prediction," *Proc. of the International Congress on Dynamics and Control of Systems*, Ottawa, Canada, Aug. 5-7, 1999.

[13] T. W. C. Williams and P. J. Ansaklis, "A unifying approach to the decoupling of linear multivariable systems," *Int. J. Control*, vol. 44, no. 1, pp. 181-201, 1986.

[14] W. A. Wolovich and P. L. Falb, "Invariants and canonical forms under dynamic compensation," *SIAM J. Control Optimization*, vol. 14, no. 6, pp. 996-1008, 1976.

SDRE ESTIMATOR PERFORMANCE
IN A HARMONIC DISTORTION PROBLEM

Edwin Engin Yaz, Yuan Gao, and Kraig Joseph Olejniczak

Department of Electrical Engineering, University of Arkansas, Fayetteville, AR 72701
E-mail: ey1@engr.uark.edu

ABSTRACT

Harmonics and inter-harmonics with random variations in magnitude and phase angles observed in electric power systems are modeled as outputs of discrete-time stochastic systems. For an active filtering of these undesirable effects, one needs to estimate the future values of these signals. In these models, the measurements are sinusoidal functions of the state variables that lead to a non-linear estimation problem. For such models, the extended Kalman filter performance is compared to that of the state-dependent Riccati equation estimator and conclusions are drawn regarding the relative performance of these filters.

INTRODUCTION

The widespread application of electronic loads containing power semiconductor devices with nonlinear voltage–current characteristics results in the increase of harmonic and inter-harmonic distortion in electric power systems [1] and [2]. Therefore, the power quality problem has been getting more and more attention by researchers.

The non-stationary stochastic state-space model for harmonics and inter-harmonics used in this work is based on the observations made in [1]-[2], regarding a combination of nonlinear effects resulting in a short-time random-like behavior. To be able to predict these effects, one needs to set up estimators. For this non-linear system, we present two estimators. One is the commonly used extended Kalman filter (EKF) and the other is the recently proposed State-Dependent Riccati Equation (SDRE) estimator [3]. In this paper, their performances are compared based on Monte-Carlo simulations.

MODEL DESCRIPTION

Consider a discrete-time signal with unknown time varying magnitude M_k, phase angle θ_k, and known and constant radian frequency ω:

$$s_k = M_k \cos(\omega\, kT + \theta_k) \tag{1}$$

where k is the discrete-time index. Let us assume that the signal is corrupted by additive noise in the measurement:

$$y_k = s_k + w_k \tag{2}$$

where w_k is assumed to be white and zero mean with known variance σ_w^2.

Although the frequency in the sinusoid (1) is known and almost constant, the magnitude and the phase angle may be unknown and time varying. A commonly used model for changes in these quantities is the first order Markov model [2]:

$$M_{k+1} = M_k + v_k^1 \tag{3}$$

$$\theta_{k+1} = \theta_k + v_k^2 \tag{4}$$

where v_k^1 and v_k^2 are zero mean noises with respective variances σ_M^2 and σ_θ^2.

If we define the state variables as

$x_k^1 = M_k,\ x_k^2 = \theta_k,$

then the signal and measurement models become:

$$\begin{aligned} x_{k+1}^1 &= x_k^1 + v_k^1 \\ x_{k+1}^2 &= x_k^2 + v_k^2 \\ y_k &= \cos(\omega kT + x_k^2)x_k^1 + w_k \end{aligned} \tag{5}$$

One can easily see that the measurement equation is nonlinear in the state variables. For a signal composed of a fundamental and (n-1) harmonics or inter-harmonics, we define 2n number of state variables to generalize this description. However, due to the summary nature of this manuscript, this is not done here. For the general model, we set up both the EKF and the SDRE for comparison purposes.

SIMULATION RESULTS

This simulation study uses the above choice of state variables where $v_k^1 \sim$ N(0, 0.05), $v_k^2 \sim$ N(0, 0.0015), $w_k \sim$ N(0, 0.01). We take $\omega T = 0.1$ and the initial conditions are $x_0^1 = 120\sqrt{2}$, $x_0^2 = \pi/6$.

Upon comparing the associated total mean square errors (mse) averaged over twenty runs of Monte Carlo simulation, we have obtained mse(EKF) = 22.86, mse(SDRE) = 1391.24, calculated by $\frac{1}{N}\sum_{k=1}^{N}(y_k - \hat{y}_k)^2$. EKF performs much better than the SDRE estimator in this particular application. The reasons for this result will be the subject of further investigation.

REFERENCES

[1] IEEE Working Group on Power System Harmonics, "Power System Harmonics: An Overview," IEEE Trans. on Power Apparatus and Systems, Vol. PAS-102, pp. 2455 – 2459, 1983.

[2] S. Varadan, A. A. Girgis, and E. B. Makram, "An FFT Perspective of Inter-Harmonics in Power Systems," IEEE Trans. on Power Delivery, pp. 386-391, 1995.

[3] P. Mracek, and C. N. D'Souza, "A New Technique for Nonlinear Estimation," Proc. of IEEE Conf. on Control Applications, Dearborn, MI, September 1996.

Stochastic Difference Equations with Time Delays

A. S. C. Sinha, Sergey E. Lyshevski,
B. R. Pidaparti and Erol Kocaoglan

Department of Electrical Engineering, Purdue University at Indianapolis
Indianapolis, Indiana 46202-5132

Abstract - The goal of the paper is the extension of familiar [1-3] results in stability analysis of linear and nonlinear discrete systems to a wider class of systems with random coefficients and delays. Our results extend the developments reported in [4-6] to discrete-time systems. The stochastic difference control systems have been studied extensively. We develop a method to analyze the mean square asymptotic stability for a class of stochastic discrete-time control systems with delays. The results are verified studying high-performance electromechanical systems with permanent-magnet synchronous machines, which are widely used as direct-drive servomotors in drives and servos [7]. Analytical and numerical results are performed to synthesize and verify the control law needed to attain the desired dynamic performances and stability in presence of the stochastic disturbances.

Extended Summary - The mathematical model of the control systems under consideration is a stochastic difference equation of the form

$$x(k+1)=\sum_{j=0}^{m}[a_j(k)+b_j(k)d\omega(k)]x(k-j)+u(k),$$

$$u(k)=c_m(k-m)x(k)+c_{m-1}(k)x(k-m+1)+\ldots+c_0(k)x(k), \tag{1}$$

$$x(k)=\varphi(k),\ \ k\in Z_0\triangleq\{-m,\ldots,0\}$$

Where $\omega(t)$ is a standard scalar Wiener process with values $\omega(k)$ at the discrete moments of time, $k=0,1,\ldots$. Hence, $E\{d\omega(k)\}=E\{\omega(k+1)-\omega(k)\}=0$ and $E\{(d\omega(k))^2\}=1$. Let us introduce the Banach space B_ρ with norm $\|\varphi\|_\rho=\sup_{k\in Z_0}\{\|\varphi\|\rho^{|k|}:0<\rho<1\}$.

We formulate the definition for asymptotic mean-square stability. Let $\{\Omega,\ \mathfrak{I},\ P\}$ be a probability space with flow of σ-algebra's, $\mathfrak{I}_t\subset\mathfrak{I}$.

Definition. Zero solution of equation (1) is called mean square stable if for every $\varepsilon>0$ there exists a $\delta>0$ such that $\mathrm{E}\{\mathrm{X}^2(\mathrm{k})\}<\varepsilon$, $k\in I\equiv\{0,1,\ldots\}$ if $\|\varphi\|^2=\sup_{n\in I_0}\mathrm{E}\{\varphi^2(\mathrm{k})\}<\delta$, where $Z_0\equiv\{-m,\ldots,0\}$. If $\lim_{n\to\infty}E\{X^2(k)\}=0$, then zero solution of (1), is mean-square stable. □

Define a vector as $X(k)=[x(k-m),\cdots,x(k-1),x(k)]^T$.

Then, the system equation (1) can be written as

$$X(k+1)=A(k)X(k)+B(k)X(k)d\omega(k)+eu(k) \tag{2}$$

The matrices $A(k)\in\mathbb{R}^{(m+1)\times(m+1)}$, $B(k)\in\mathbb{R}^{(m+1)\times(m+1)}$, $e=\begin{bmatrix}0 & 0 & \cdots & 0 & 1\end{bmatrix}^T\in\mathbb{R}^{m+1}$ are

$$A(k)=\begin{bmatrix}0 & 1 & 0 & \cdots & 0\\ 0 & 0 & 1 & \cdots & 0\\ \cdots & \cdots & \cdots & \ddots & \cdots\\ 0 & 0 & 0 & \cdots & 1\\ a_m(k) & a_{m-1}(k) & a_{m-2}(k) & \cdots & a_0(k)\end{bmatrix},\ B(k)=\begin{bmatrix}0 & 1 & 0 & \cdots & 0\\ 0 & 0 & 1 & \cdots & 0\\ \cdots & \cdots & \cdots & \ddots & \cdots\\ 0 & 0 & 0 & \cdots & 1\\ b_m(k) & b_{m-1}(k) & b_{m-2}(k) & \cdots & b_0(k)\end{bmatrix}$$

The closed-loop system has the form

$$X(k+1) = A_\mu(k)X(k) + B(k)X(k)d\omega(k) \tag{3}$$

Where $$A_\mu(k) \triangleq A(k) + \mu(k)C(k); \quad 0 \le \mu(k) \le 1 \tag{4}$$

and $$C(k) \triangleq \begin{bmatrix} 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & \cdots & 0 & 0 \\ \cdots & \cdots & \ddots & 0 & 0 \\ 0 & 0 & \cdots & 0 & 0 \\ c_m(k) & c_{m-1}(k) & \cdots & c_1(k) & c_0(k) \end{bmatrix} \tag{5}$$

Define $\Phi(k,k_0) \triangleq A(k-1)A(k-2)\cdots A(k_0)$. Let us estimate $\|\Phi(k,k_0)\|$. We have obtained the following result for the linear system

$$x(k+1) = A_\mu(k)x(k) \tag{6}$$

Theorem 1. Let there exist positive constants ρ and ρ_0 such that

(i) $\sup\limits_{k\ge 0}\|A_\mu(k)\| \le \rho$, and (ii) $\sum\limits_{n=k_0}^{k} n\|\Phi(k,n)\| \le \rho_0,\ \rho_0 > 0$,

then, there exist positive constants γ, α and M ($\gamma < 1,\ \alpha > 0\ and\ M > 0$) such that for any $0 \le k_0 \le k < \infty$ the fundamental matrix of the system (6) satisfies the inequality

$\|\Phi(k,k_0)\| \le M\gamma^{k-k_0}$. □

The selection of the feedback gain matrix *C*(*k*) allows us to guarantee the criterion (i).

Theorem 2. Assume that for closed-loop system (6) the norm of matrices $A_\mu(k)$ and *B*(*k*) satisfy the following bounds $\|A_\mu(k)\| \le \rho\ and\ \|B(k)\| \le \sigma$. Let the conditions of Theorem 1 hold. That is, there exists a Liapunov function $V(X(k))$ such that $E[\Delta V(\cdot)] = E[V(X(k+1)) - V(X(k))] \le 0$. Then, the solutions of the closed-loop system (3) are mean-square stable. □

Example. Permanent-magnet synchronous motors in the rotor reference frame are modeled as [7]

$$\frac{di_{qs}^r}{dt} = -\frac{r_s}{L_{ss}}i_{qs}^r - \frac{\psi_m}{L_{ss}}\omega_r - i_{ds}^r\omega_r + \frac{1}{L_{ss}}u_{qs}^r,\quad \frac{di_{ds}^r}{dt} = -\frac{r_s}{L_{ss}}i_{ds}^r + i_{qs}^r\omega_r + \frac{1}{L_{ss}}u_{ds}^r,$$

$$\frac{d\omega_r}{dt} = \frac{3P^2\psi_m}{8J}i_{qs}^r - \frac{B_m}{J}\omega_r - \frac{P}{2J}T_L,$$

where i_{qs}^r, i_{ds}^r and u_{qs}^r, u_{ds}^r are the *quadrature* and *direct* currents and voltages; ω_r is the electrical angular velocity; T_L is the load; r_s is the stator resistance; L_{ls}, L_{ms} and L_{ss} are the leakage, magnetizing, and self inductances, $L_{ss} = L_{ls} + \frac{3}{2}L_{ms}$; ψ_m is the flux; B_m is the viscous friction coefficient; *J* is the moment of inertia; *P* is the number of poles.

Denoting the sampling period as T_s and using the forward Euler approximation we have

$$\begin{aligned} i_{qs}^r(n+1) &= -\frac{r_s(T_s+1)}{L_{ss}}i_{qs}^r(n) - \frac{\psi_m T_s}{L_{ss}}\omega_r(n) - T_s i_{ds}^r(n)\omega_r(n) + \frac{T_s}{L_{ss}}u_{qs}^r(n), \\ i_{ds}^r(n+1) &= -\frac{r_s(T_s+1)}{L_{ss}}i_{ds}^r(n) + T_s i_{qs}^r(n)\omega_r(n) + \frac{T_s}{L_{ss}}u_{ds}^r(n), \\ \omega_r(n+1) &= \frac{3P^2\psi_m T_s}{8J}i_{qs}^r(n) - \frac{B_m(T_s+1)}{J}\omega_r(n) - \frac{PT_s}{2J}T_L(n-1). \end{aligned} \tag{7}$$

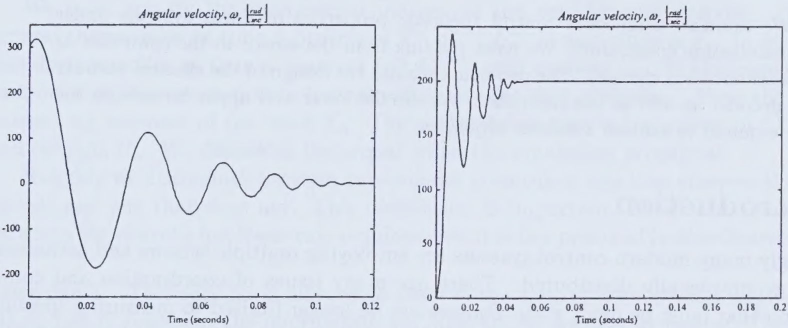
We use the Kollmorgen motor H-232. The motor parameters are:
$r_s(\cdot) \in [0.5_{T=20^\circ C}\ 0.75_{T=130^\circ C}]$ ohm, L_{ss}=0.009 H, L_{ls}=0.001 H, $\psi_m(\cdot) \in [0.07_{T=20^\circ C}\ 0.055_{T=130^\circ C}]$ V-sec/rad, $\psi_m(\cdot) \in [0.07_{T=20^\circ C}\ 0.055_{T=130^\circ C}]$ N-m/A, B_m=0.000013 N-m-sec/rad and J=0.0001 kg-m^2.
The controller is found using the design procedure reported. In particular,

$$u_{qs}^r(n) = -\left(0.87 i_{qs}^r(n) + 0.49 i_{ds}^r(n) + 0.61 \omega_r(n)\right). \tag{8}$$

The asymptotic stability of the closed-loop system (7)-(8) is guaranteed by the controller (8), which implies that conditions (i) and (ii) of Theorem 1 hold. The simulations were performed, and the evolution of the angular velocity (if the initial conditions of the state variables are 10 A, 5 A and 300 rad/sec) is illustrated in Figure 1.a.
The tracking controller is designed as

$$u_{qs}^r(n) = \left(15.7 e(n) - 0.87 i_{qs}^r(n) - 0.49 i_{ds}^r(n) - 0.61 \omega_r(n)\right),\ e(n) = r(n) - \omega_r(n). \tag{9}$$

Figure 1.b documents the motor dynamics if the desired angular velocity is assigned to be 200 rad/sec. In modeling, the tracking controller (9) was implemented.



a *b*

Figure 1. Transient dynamics of the motor.

References

[1] Sinha, A. S. C. (1997), Control of dissynchronized systems of difference equations with delay, *The 35th Annual Allerton Conference on Communication, Ccontrol and Computing,* University of Illinois, IL.

[2] Kolmanovskii, V.B., and Myshis, A.D. (1992), *Applied Theory of Functional Differential Equations*, Kluwer Academic Publishers, Boston.

[3] Ladde, G. S. (1981), Stochastic stability analysis of model ecosystems with time-delay, *Differential Equations and Applications in Biology, Epidemics, and Population Problems,* Edited by S. N. Busenberg and K. Cook, Academic Press, NY.

[4] Lakshmikantham, V. and Trigiante, D. (1988). Theory of difference equations: numerical methods and applications, *Mathematics in Science and Engineering,* vol. 181, Academic Press, NY.

[5] Srichander, R., and Walker, B. K. (1993), Stochastic stability analysis for continuous-time fault tolerant control systems, *International J. of Control,* vol. 57, 433-452.

[6] Freeman, R. A. and P. V. Kokotovic (1995). Optimal nonlinear controllers for feedback linearizable systems. *Proc. American Control Conference*, Seattle, WA, vol. 4, pp. 2722-2726.

[7] Lyshevski, S. E. (1999), *Electromechanical Systems, Electric Machines, and Applied Mechatronics*, CRC Press, FL.

Control Under Communication Constraints*

Sekhar Tatikonda and Sanjoy Mitter
Laboratory for Information and Decision Systems
Massachusetts Institute of Technology
Cambridge, MA 02139
Corresponding author: tatikond@mit.edu

Abstract

We examine traditional control theoretic properties of linear systems under communication constraints. We treat the link from the sensor to the controller as a communication channel. The problem includes the design of the channel encoder and decoder as well as the controller. We discuss lower and upper bounds on the rate required to achieve a control objective.

1 Introduction

Increasingly many modern control systems are employing multiple sensors and actuators that are geographically distributed. There are many issues of coordination and communication that must be dealt with. Sensors are no longer limited to measuring specific physical properties of interest and the raw data that they capture can easily be megabytes per second. It becomes important then to determine what the sensors should transmit to the controller and what channel rates are required to achieve a specified performance.

In this paper we formulate a control problem with a communication channel connecting the plant to the controller. This channel can transmit R bits noiselessly per time step. Our task is to design the encoder and decoder along with the controller to achieve some control objective. For a given control objective we provide upper and lower bounds on the rate required to achieve it. In many cases these bounds are tight. In doing so we characterize the "information complexity" of different control objectives.

We view this problem as a partially observed control problem where we design the observation equation subject to the rate constraint. In general, communication constrained control problems of this sort are very hard because they depend crucially on two difficult subproblems: control of partially observed processes and real-time information theory.

Throughout we assume the control objective of interest is achievable via full state observation. Clearly if an objective cannot be achieved via full state observation it cannot be achieved under the rate constraint. Conversely if a control objective can be achieved under a rate constraint R then it can be achieved under full state observation. This simple observation is a generalization of the data-processing inequality in information theory.

---

[*] This research is supported U.S. Army Grant PAAL03-92-G-0115, Center for Intelligent Control Systems.

We round out this section with a discussion of encoders and decoders. In section 2 we discuss upper bounds on the rate required for a variety of different objectives. In section 3 we discuss lower bounds on the rate and mention when the bounds are tight. We then conclude in section 4. We leave detailed proofs to the complete paper. [Tat]

We consider the following system:

$$X_{n+1} = AX_n + BU_n \quad n \geq 0 \tag{1}$$

where $\{X_n\}$ is a $I\!R^d$-valued state process and $\{U_n\}$ is a $I\!R^m$-valued control process. The initial position $X_0 \in \Lambda_0$ where $\Lambda_0 \subseteq I\!R^d$. A, B are matrices of suitable dimensions.

1.1 The Encoder and Decoder

At each time step the encoder transmits one symbol from the channel alphabet Σ where $|\Sigma| = 2^R$. The encoder and decoder may have state and thus must be synchronized to allow the decoder to decode the received symbol σ properly.

We must specify the information pattern of the encoder and decoder. [Wit] In general the encoder at time n observes $(X^n, \sigma^{n-1}, U^{n-1})$ and outputs σ_n. In general the decoder/controller at time n observes (σ^n, U^{n-1}) and outputs U_n. In this paper we will assume a separation structure between the decoder and controller. Here the decoder outputs an estimate of the state $\hat{X}_n$. The controller at time n has access to $(\hat{X}^n, U^{n-1})$ and outputs U_n. We discuss in the sequel when the separation is optimal.

Roughly we distinguish between two encoder structures: one that observes the control signals and one that does not. This distinction is important. While the encoder with access to the controls has lower rate requirements it is less practical in distributed settings.

Encoder 1 This encoder is allowed to observe the control signals being applied to the plant. Let I_n represent the information available to the encoder at time n. The encoder evolves according to (in an abuse of notation):

$$I_{n+1} = \{I_n, X_n, U_n, \sigma_n\}$$
$$\sigma_n = \mathcal{E}_n(I_n, X_n).$$

with $I_0 = \{X_0\}$. Similarly the decoder evolves according to:

$$J_{n+1} = \{J_n, U_n, \sigma_n\}$$
$$\hat{X}_n = \mathcal{D}_n(J_n, \sigma_n).$$

with $J_0 = \{\hat{X}_0\}$ and J_n is the decoder information. Note that the optimal action for the encoder in this case is to compute the optimal control and then quantize it appropriately.

Encoder 2 This encoder does not have access to the control signals but may know the control law. This might be the case where the sensor is geographically separate from the plant. The encoder evolves according to:

$$I_{n+1} = \{I_n, X_n, \sigma_n\}$$
$$\sigma_n = \mathcal{E}_n(I_n, X_n).$$

with $I_0 = \{X_0\}$. The decoder evolves the same was as for decoder 1. In this case it is less obvious what the encoder should transmit.

For both encoders the encoding map $\mathcal{E}_n$ based on it's information I_n or J_n selects one quantizer parameterized as follows:

Definition 1.1 *A $(C, \underline{R}, \underline{L}, \Phi)$-quantizer with $C \in \mathbb{R}^d$, $\underline{R}, \underline{L} \in \mathbb{R}^{d,+}$, and Φ an invertible matrix representing a coordinate change, partitions the region $\Lambda = \{X \in \mathbb{R}^d : \Phi(X-C) \in \{[-L_1, L_1] \times \ldots \times [-L_d, L_d]\}$ into boxes with 2^{R_i} equally spaced partitions in each coordinate direction. Each of the 2^R boxes is represented by an element $\sigma \in \Sigma$. Upon observing X_n the $(C, \underline{R}, \underline{L}, \Phi)$-quantizer subtracts off C, applies the coordinate transform Φ, determines which box it falls into, and then transmits the σ representing that box. If X falls outside the region Λ then the quantizer transmits one symbol representing an overflow. Thus we have $2^R + 1$ symbols.*

2 Upper bounds

We provide upper bounds on the rate by explicitly describing encoder/decoder and controller schemes.

2.1 Observability

An encoding scheme applied to system (1) is asymptotically observable if $\lim_{n\to\infty} \|e_n\|_\infty = 0$ where $e_n = X_n - \hat{X}_n$. First we treat the autonomous case (i.e. assume that the control $U_n = 0$.) For this case both encoder 1 and 2 are the same.

Theorem 2.1 *For the autonomous case, a sufficient condition for both encoder 1 and 2 to achieve asymptotic observability is $R > \sum_{\lambda(A)} \max\{0, \log \lambda(A)\}$.*

Proof: Assume first that the initial uncertainty is a bounded set $\Lambda_0 \subset \{X : \|X\|_2 \leq L\}$. Let Φ diagonalize A. At time zero choose a $(C(0), \underline{R}, \underline{L}(0), \Phi)$-quantizer where $C(0)$ is the origin, $L_i(0) = L \; \forall i$, and $R_i > max\{0, \log \lambda_i(A)\}$. Update the quantizer as follows: $C(n+1) = C(n) - \hat{X}_n$ and $L_i(n+1) = L_i(n) 2^{(\log(\lambda_i(A)) - R_i)}$. Let $\hat{X}_n$ be the centroid of the region represented by the received symbol σ_n. The error can then be shown $\|e_n\|_\infty \leq L\, 2^{-n \min_i (R_i - \log(\lambda_i(A)))}$. Thus we can characterize the rate of exponential convergence.

For the case where the initial uncertainty, Λ_0, is an unbounded set one must first "capture" the state in the quantizer domain. Specifically let $(C(0), \underline{R}, \underline{L}(0), \Phi)$-quantizer where $C(0)$ is the origin, $L_i(0) = L \; \forall i$ for some arbitrary L. If upon observing X_n the quantizer at time transmits an overflow symbol then update the quantizer as follows: $C(n+1) = C(n)$ and $L_i(n+1) = L_i(n) 2^{R_i}$. Since the L's are growing faster than the state eventually the quantizer will capture the state. At this point proceed as we did previously. □

For the nonautonomous case there is a distinction between the two encoders. For encoder 1 we can describe a scheme that achieves asymptotic observability independent of the control signals applied. For encoder 2 this is no longer the case.

Theorem 2.2 *For the nonautonomous case, a sufficient condition for encoder 1 to achieve asymptotic observability is $R > \sum_{\lambda(A)} \max\{0, \log \lambda(A)\}$.*

Proof: The encoder observes the control signal and thus can subtract out it's effects from the observation of X. It then transmits a quantization of the innovation as before. □

For encoder 2 one must place some restrictions on the control signals allowed. We discuss this in section 2.2. If there are no restrictions on the control sequences then we get:

Theorem 2.3 *For the nonautonomous case, one cannot find a finite rate scheme for encoder 2 such that we achieve asymptotic observability under all control sequences.*

Proof: Because we are allowing any control sequence the state can be arbitrarily far from the origin. Since the encoder does not know the control signal it cannot hope to track the state. □.

We can extend the results to systems with bounded disturbances. Consider

$$X_{n+1} = AX_n + BU_n + W_n \quad n \geq 0 \tag{2}$$

where $\|W_n\|_2 \leq D$.

Theorem 2.4 *For encoder 1,* $R > \sum_{\lambda(A)} \max\{0, \log \lambda(A)\}$ *is sufficient to drive* $\lim_{n\to\infty} \|e_n\|_\infty \leq D$.

Proof: Follow Theorem 2.1 with $L_i(n+1) = L_i(n)2^{(\log(\lambda_i(A))-R_i)} + D2^{-R_i}$. □

Note that the disturbances do not effect the rate requirement. But we can never hope to drive the observation error to zero.

2.2 Stability and Controllability

Assume (A, B) are stabilizable. An encoder and controller scheme applied to system (1) is asymptotically stable if $\lim_{n\to\infty} \|X_n\|_\infty = 0$.

Theorem 2.5 *A sufficient condition for a scheme using encoder 1 to achieve asymptotic stability is* $R > \sum_{\lambda(A)} \max\{0, \log \lambda(A)\}$.

Proof: The system is asymptotically observable under any control sequence. A system that is both stabilizable and asymptotically observable is output stabilizable. [Vid] □

Encoder 2 is not allowed to see the controls. But it is assumed to know the control objective.

Theorem 2.6 *A sufficient condition for encoder 2 to achieve asymptotic stability is* $R > \sum_{\lambda(A)} \max\{0, \log \lambda(A)\}$.

Proof: First assume Λ_0 is bounded by L as before. Let Φ diagonalize A. Assume the controller is the full state minimum variance controller applied to the state estimate. Let $C(n) = 0$ for all n. And let $L_i(0) = L$ and $L_i(n+1) = L_i(n)2^{(\log(\lambda_i(A))-R_i)}$. One can show that under the minimum variance control law that $X_n \in \{X : \Phi X \in \{[-L_1(n), L_1(n)] \times \ldots \times [-L_d(n), L_d(n)]\}$. If Λ_0 is unbounded then apply the zero control and expand the quantizer domains until we "capture" the state as in theorem 2.1. □

The main idea here is that the encoder has a schedule of X_n. It knows that it will lie in a neighborhood of the origin. And if the rate is large enough then under the minimum variance control law this neighborhood will shrink. We can replace knowledge of the actual control signals with that of knowledge of the control policy and attendant convergence schedule.

Assume (A, B) are controllable. An encoder and controller scheme applied to system (1) is asymptotically controllable if for any X_0 and Y one can drive the system $\lim_{n\to\infty} \|X_n - Y\|_\infty = 0$.

Theorem 2.7 *A sufficient condition for encoder 1 to achieve controllability is* $R > \sum_{\lambda(A)} \max\{0, \log \lambda(A)\}$.

Proof: The system is asymptotically observable under any control sequence. A system that is both controllable and asymptotically observable is asymptotically controllable.

Encoder 2 takes a bit more work.

Theorem 2.8 *A sufficient condition for encoder 2 to achieve controllability in a* $R > \sum_{\lambda(A)} \max\{0, \log \lambda(A)\}$.

Proof: The basic idea is to first drive the state to a neighborhood of the origin. This can be done with the rate mentioned. Then we define a new set $\tilde{\Lambda}$ that is large enough to contain both this neighborhood of the origin and Y. We then treat Y as the origin of a new stability problem and drive the state to Y as we did in theorem 2.6. □

Note that in this proof the control has a dual nature. First it is used to observe the state by driving the state to the origin. Then it is used to control the state by driving the state to Y.

2.3 Quadratic Cost

So far we have seen that the rates required for both encoder 1 and encoder 2 for the problems of observability, stabilizability, and controllability are the same. Though the complexity of the encoder 2 schemes are much greater than that of the encoder 1 schemes. Here we show that for the quadratic cost criterion the rate required for the encoder 2 scheme is much greater than that of the encoder 1 scheme.

Our objective function is the quadratic cost:

$$E\left(X_N' Q X_N + \sum_{i=0}^{N-1} (X_k' Q X_k + U_k' T U_k)\right) \tag{3}$$

Where $Q \in \mathbb{R}^{d\times d}$ is symmetric positive semidefinite, $T \in \mathbb{R}^{m\times m}$ is symmetric positive definite, and $(A, Q^{\frac{1}{2}})$ is observable. Further Λ_0 is assumed to be bounded. These are the standard LQ assumptions and can be found, for instance, in [Ber].

Because encoder 1 observes both the state and the control sequence it satisfies a condition called equi-memory. [TSM] Basically this states that the encoder knows the decoder's estimate of the state. Further it is shown in [TSM] that the optimal encoder/decoder/controller scheme under equi-memory is a separation scheme where the controller is the certainty equivalent controller. The job of the encoder/decoder then is to minimize the state estimation error.

Specifically let K_n solve the Riccati equation

$$K_{n-1} = A'(K_n - QB(B'K_nB + T)^{-1}BK_n)A + Q.$$

Then the cost can be shown to separate into two pieces: $X_0'K_0X_0 + D_{seq}(R)$ where $D_{seq}(R)$ is the sequential distortion rate function. [TSM] The first cost is the full state observation cost. The second cost is the cost due to the estimation error induced by quantization.

Theorem 2.9 $R_{seq}(D) = \infty$ *if* $R \leq \sum_{\lambda(A)} \max\{0, \log \lambda_i(A)\}$ *and goes to zero as* $R \to \infty$.

Proof: See [TSM]. □

In summary

Theorem 2.10 *For encoder 1, if the* $R \geq \sum_{\lambda(A)} \max\{0, \log \lambda_i(A)\}$ *then the optimal scheme is a separation scheme and the infinite horizon cost is finite.*

For encoder 2 the problem becomes more difficult for we no longer have the equi-memory condition. The encoder does not know the decoder's state estimate because it does not have access to the control signals. Once again we treat the problem by providing a convergence schedule to encoder 2.

Assume that the certainty equivalent controller for the cost (3) is $U_n = KX_n$. The encoder 2 knows K and thus given a rate knows the convergence schedule. In this case the rate depends on the closed loop eigenvalues.

Theorem 2.11 *For encoder 2 a sufficient condition for the infinite horizon cost to be finite is* $R > \sum_{\lambda_i(A+BK)} \log \frac{\|BK\|_\infty}{1-\lambda_i(A+BK)}$.

Proof: The convergence schedule decreases with respect to the closed loop eigenvalues $A + BK$. It can be shown that the rate condition is sufficient to ensure that under the control K applied to the state estimate the state never leaves this convergence schedule. See [Tat]. □

Note that as the closed loop eigenvalues get close to the unit circle this bound increases to infinity. Thus large control costs will lead to large rates. In practice one would want to balance these costs.

3 Lower bounds

We provide lower bounds on the rate by using counting arguments.

3.1 Observability

Theorem 3.1 *In the autonomous case, a necessary condition for both encoder 1 and 2 to achieve asymptotic observability* $R > \sum_{\lambda(A)} \max\{0, \log \lambda(A)\}$.

Proof: Assume that A is diagonal. Assume first that the initial uncertainty is a bounded set $\Lambda_0 \subset \{X : \|X\|_\infty \leq L\}$. At time n the set of reachable states is $\{X : -L\lambda_i^n \leq X_i \leq L\lambda_i^n \; i = 1, .., d\}$. In that time n we can transmit $2^{(n+1)R}$ symbols. Thus at time n the finest resolution we can describe X_n in the infinity norm sense is

$$\frac{L^u(\lambda_1 ... \lambda_u)^n}{2^{(n+1)R}} = \frac{L^u}{2^R} 2^{-n(\sum_\lambda R_i - \log \lambda_i)}$$

where $\lambda_1, ..., \lambda_u$ are the unstable eigenvalues. Clearly this goes to zero only if $R > \sum_{\lambda(A)} \max\{0, \log \lambda(A)\}$. Further note that since this rate is required for the bounded initial uncertainty it certainly must also hold for the unbounded initial uncertainty. □

Theorem 3.2 *For the nonautonomous case, a necessary condition for encoder 1 to achieve asymptotic observability is* $R > \sum_{\lambda(A)} \max\{0, \log \lambda(A)\}$.

Proof: The encoder has access to the controls and thus can subtract out their effect. □

As we discussed in the last section encoder 2 we cannot achieve asymptotically observability without some sense of what controls the system will be using.

3.2 Stability and Controllability

Theorem 3.3 *A necessary condition for a scheme using encoder 1 to achieve asymptotic stability is* $R > \sum_{\lambda(A)} \max\{0, \log \lambda(A)\}$.

Proof: Let $\Lambda_0 \subset \{X : \|X\|_\infty \leq L\}$. For a given control sequence $U_0, U_1, ..., U_{n-1}$ we have

$$X_n = A^n X_0 + \sum_{i=0}^{n-1} A^{n-1-i} B U_i$$

Define for some $\epsilon > 0$ the balls Γ parameterized by the control sequence $U_0, ..., U_{n-1}$

$$\Gamma_{U_0^{n-1}} = \{X_0 : \|X_n\|_\infty < \epsilon\}.$$

We now lower bound the minimum number of Γ balls required to cover Λ_0. This is $\lambda_1^n(A) \times, ..., \times \lambda_u^n(A)(\frac{L}{\epsilon})$. (Where $\lambda_1, ..., \lambda_u$ are the unstable eigenvalues.) The largest number of different control sequences we may apply in time n is $2^{n+1}R$. Thus the rate $R > \sum_{\lambda(A)} \max\{0, \log \lambda_i(A)\} + \frac{1}{n} \log \frac{L}{\epsilon}$. □

Theorem 3.4 *A necessary condition for a scheme using encoder 2 to achieve asymptotic stability is* $R > \sum_{\lambda(A)} \max\{0, \log \lambda(A)\}$.

Proof: Clearly if this rate is required for encoder 1 then it must be required for encoder 2. □

Theorem 3.5 *A necessary condition for a scheme using encoder 1 to achieve asymptotic controllability is* $R > \sum_{\lambda(A)} \max\{0, \log \lambda(A)\}$.

Proof: The proof is similar to theorem 3.3 except one asks how many balls $\Gamma_{U_0^{n-1}} = \{X_0 : \|X_n - Y\|_\infty < \epsilon\}$ are needed to cover Λ_0. □

Theorem 3.6 *A necessary condition for a scheme using encoder 2 to achieve asymptotic controllability is* $R > \sum_{\lambda(A)} \max\{0, \log \lambda(A)\}$.

Proof: Clearly if this rate is required for encoder 1 then it must be required for encoder 2. □

3.3 Quadratic Cost

Theorem 3.7 *A necessary condition for encoder 1 to ensure that the infinite horizon quadratic cost (3) is finite is* $R > \sum_{\lambda(A)} \max\{0, \log \lambda(A)\}$.

Proof: Follows from theorem 2.9. □.

For encoder 2 one must use a more complicated counting argument to lower bound the rate required to ensure a finite cost. Fix $C < \infty$. Define $\Gamma_{U_0^{n-1}} = \{X_0 : X_n' Q X_n + \sum_{i=0}^{N-1} (X_k' Q X_k + U_k' T U_k) \leq C\}$. The sets Γ are convex. In principle one can ask how many such ellipses are needed to cover Λ_0. But it is not obvious how to incorporate the provision that encoder 2 does not observe the controls. It is an open question to determine a lower bound for encoder 2 for the quadratic cost problem.

4 Conclusion

In this note we have examined traditional control objectives under communication constraints. In many cases we are able to get tight bounds on the minimum rate required to achieve these objectives. We noted that there is a large difference in rate requirements if the encoder is allowed to observe the control or not. This difference manifests itself especially when looking at performance criterion.

Extensions to the stochastic setting can be found in [BM], [BB], and [TSM]. Extensions to the nonlinear control setting and distributed setting can be found in [Tat].

References

[BB] R. Bansal and T. Basar, "Simultaneous Design of Measurement and Control Strategies for Stochastic Systems with Feedback." Automatica, Volume 25, No. 5, pp 679-694, 1989.

[Ber] D. Bertsekas, *Dynamic Programming and Optimal Control.* Belmont, MA: Athena Scientific, 1995.

[BM] V. Borkar and S. Mitter, "LQG Control with Communication Constraints." LIDS report number LIDS-P-2326, M.I.T. December 1995.

[TSM] S. Tatikonda, A. Sahai, and S. Mitter, "Control of LQG Systems Under Communication Constraints," forthcoming paper.

[Tat] S. Tatikonda; "Control Under Communication Constraints," MIT Ph.D. thesis in progress.

[Vid] M. Vidyasagar, "On the Stabilization of Nonlinear Systems Using State Detection." IEEE Trans. on Automatic Control, Vol. 25, No. 3, pp. 504-509, June 1980.

[Wit] H. Witsenhausen, "Separation of Estimation and Control for Discrete Time Systems." Proceedings of the IEEE, Volume 59, No. 11, November 1971.

Space-time Fading Channel Estimation in Unknown Spatially Correlated Noise*

Aleksandar Dogandžić and Arye Nehorai

EECS Department (M/C 154)
University of Illinois at Chicago, Chicago, IL 60607
{adogandz, nehorai}@eecs.uic.edu

Abstract

We present maximum likelihood (ML) methods for space-time fading channel estimation using an antenna array in spatially correlated noise with unknown covariance. The received signal is modeled as a linear combination of discrete multipath-delayed and Doppler-shifted copies of the transmitted waveform. We consider structured and unstructured array response models, and derive the Cramér-Rao bound (CRB) for the unknown direction and signal parameters. We also propose coherent matched-filter and decision-feedback receivers which account for spatially correlated interference.

1 Introduction

To combat fading, the parameters of a communication channel need to be determined. In this paper, we propose algorithms for space-time estimation of deterministic time-varying multipath channels in the presence of spatially correlated noise with unknown covariance, with application to wireless communications [1]–[6] and radar [7], [8]. Both structured and unstructured array responses are considered, and the signal is modeled as a linear combination of basis functions which account for multipath and Doppler diversity, see Section 2. In Sections 3 and 4, we present maximum likelihood (ML) estimation algorithms for structured and unstructured array response models, respectively. In Section 5 we compute the Cramér-Rao bound (CRB) for the unknown direction of arrival (DOA) and signal (Doppler shift and time delay) parameters. We show that the bounds on the DOA and signal parameters are uncoupled. In Section 6 we incorporate both the channel and spatial noise covariance estimates in the design of communication receivers.

2 Models

Suppose an m-element antenna array receives several scaled, time delayed, and Doppler-shifted copies of a known complex baseband signal $s(t)$. Then, the $m \times 1$ vector signal received by the array at time t becomes

$$\boldsymbol{y}(t) = \sum_{d=1}^{D} \boldsymbol{a}(\theta_d) \cdot x_d \cdot \exp(j\omega_{\mathrm{D}d} t) s(t-\tau_d) + \boldsymbol{e}(t), \quad t = 1, \ldots, N, \tag{2.1}$$

where θ_d, x_d, τ_d, and $\omega_{\mathrm{D}d}$ are the DOA, complex amplitude, time delay, and Doppler shift for the d-th path ($d = 1, 2, \ldots, D$), and $\boldsymbol{e}(t)$ is additive noise. In (2.1) it is assumed that

---

*This work was supported by the Air Force Office of Scientific Research under Grants F49620-97-1-0481 and F49620-99-1-0067, the National Science Foundation under Grant MIP-9615590, and the Office of Naval Research under Grant N00014-98-1-0542.

the propagation time of the signal across the array is much smaller than the reciprocal of the signal bandwidth.

The model in (2.1) contains a large number of parameters, especially when the number of paths D is large. A way to solve this is to discretize the time delays τ_d and Doppler shifts $\omega_{\mathrm{D}d}$ by using the following basis function expansion:

$$y(t) = A(\boldsymbol{\theta})X\boldsymbol{\phi}(t,\boldsymbol{\eta}) + \boldsymbol{e}(t),\ t = 1,\dots,N. \tag{2.2}$$

Here, $A(\boldsymbol{\theta})$ is an $m \times r$ parametrization of the array response with full rank r, and X is an $r \times l$ matrix of unknown coefficients for the functional representation described by an $l \times 1$ vector of basis functions

$$\boldsymbol{\phi}(t,\boldsymbol{\eta}) = [u_{0,-K}(t,\boldsymbol{\eta}), u_{0,-K+1}(t,\boldsymbol{\eta}), \dots, u_{0,K}(t,\boldsymbol{\eta}), u_{1,-K}(t,\boldsymbol{\eta}), \dots, u_{P-1,K}(t,\boldsymbol{\eta})]^T. \tag{2.3}$$

In the above formulation, $l = P \cdot (2K+1)$, $u_{p,k}(t,\boldsymbol{\eta}) = s(t - \tau - p\Delta\tau)\exp[j(\omega_{\mathrm{D}} + k\Delta\omega_{\mathrm{D}})t]$ for $p = 0, 1, \dots, P-1$ and $k = -K, \dots, K$, τ and ω_{D} are the "global" time delay and Doppler shift, respectively, and $\Delta\tau$ and $\Delta\omega_{\mathrm{D}}$ are the corresponding intervals with which the time delay and Doppler shift are discretized. The choice of $\Delta\tau$ and $\Delta\omega_{\mathrm{D}}$ depends on the bandwidth and duration of the signal $s(t)$; thus, these quantities are known *a priori* [2]. Therefore, τ and ω_{D} are the unknown basis function parameters to be estimated, forming $\boldsymbol{\eta} = [\omega_{\mathrm{D}}, \tau]^T$. The basis functions similar to (2.3) have been introduced in [1], [2] (see also references therein) to model fast-fading, frequency-selective communication channels. Unlike (2.3), it is assumed in [1], [2] that the channel is perfectly synchronized to the "global" delay and Doppler shift, which are set to zero.

Note that the model in (2.1) is also a special case of the basis function formulation in (2.2), with $A(\boldsymbol{\theta}) = [\boldsymbol{a}(\theta_1) \cdots \boldsymbol{a}(\theta_{\mathrm{D}})]$, $X = \operatorname{diag}(x_1, x_2, \dots, x_D)$, $\boldsymbol{\phi}(t,\boldsymbol{\eta}) = [\exp(j\omega_{\mathrm{D}1}t)s(t - \tau_1), \dots, \exp(j\omega_{\mathrm{D}D}t)s(t - \tau_D)]^T$ for $t = 1, \dots, N$, and $\boldsymbol{\eta} = [\omega_{\mathrm{D}1}, \omega_{\mathrm{D}2}, \dots, \omega_{\mathrm{D}D}, \tau_1, \tau_2, \dots, \tau_D]^T$.

The noise term $\boldsymbol{e}(t)$ models interference due to receiver noise, clutter (in radar) or co-channel interference (CCI) in wireless communication channels. When dominated by clutter or CCI, $\boldsymbol{e}(t)$ is both temporally and spatially correlated. Throughout this paper we assume that $\boldsymbol{e}(t)$ is zero-mean Gaussian, temporally uncorrelated and spatially correlated with unknown positive definite covariance Σ, see also [3]–[5]. A similar noise model is used in [6] for estimating time delays in direct-sequence code division multiple access (DS-CDMA) systems, and in [7] to model clutter. In [7] temporal correlation was eliminated by pre-processing the data with the pre-whitening vector autoregressive (VAR) filter, whose parameters were estimated from the target-free data. In [6] the fast Fourier transform (FFT) was applied to the data, exploiting the fact that the noise will be asymptotically uncorrelated on the frequency grid. In this paper, we ignore temporal correlation, i.e. the data has already been pre-whitened, if necessary.

Define the vector of channel parameters $\boldsymbol{\rho} = [\operatorname{vec}(\operatorname{Re}\{X\})^T, \operatorname{vec}(\operatorname{Im}\{X\})^T, \boldsymbol{\theta}^T, \boldsymbol{\eta}^T]^T$ and the mean (i.e. noiseless) signal vector at time t as $\boldsymbol{\mu}(t,\boldsymbol{\rho}) = A(\boldsymbol{\theta})X\boldsymbol{\phi}(t,\boldsymbol{\eta})$. Our goal is to estimate $\boldsymbol{\rho}$ and the noise covariance Σ. For communication applications in particular, we are interested in synchronizing the receiver to the global delay and Doppler shift (defined by $\boldsymbol{\eta}$) and estimating the channel matrix $H = A(\boldsymbol{\theta})X$. Then, we will use the obtained estimates of H and Σ for receiver design, see Section 6.

For the structured array model in (2.2), we need a suitable parametrization of the array response matrix $A(\boldsymbol{\theta})$. For example, we may discretize the directions of arrival around a "global" DOA [2], choose $A(\boldsymbol{\theta}) = [\boldsymbol{a}(\theta_1) \cdots \boldsymbol{a}(\theta_r)]$ where $\boldsymbol{\theta} = [\theta_1, \dots, \theta_r]$, or use a distributed source model. If, however, we are not interested in DOA estimation, or do not have a suitable parametrization for the array response matrix, we may use the unstructured array model described below.

In the unstructured array response model the whole array response vector is assumed to be unknown [9], [10]. This model is robust compared with an incorrectly structured array model. Inaccuracies of the calibrated structured array may be due to changes in antenna locations, temperature and the surrounding environment, as well as gain and phase errors, mutual coupling, quantization and interpolation errors, etc. In some cases these inaccuracies can lead to a significant loss in estimation and detection performance, which motivates the application of the unstructured array model. Also, the parametrization in (2.1) involves the nonlinear dependence of the data model on $\boldsymbol{\theta}$, which needs to be estimated using a nonlinear optimization procedure. The unstructured array model can be written as

$$\boldsymbol{y}(t) = H\boldsymbol{\phi}(t,\boldsymbol{\eta}) + \boldsymbol{e}(t),\ t = 1,\ldots,N, \tag{2.4}$$

with the same noise assumptions as for the structured model. Here H is the channel response matrix, parametrized as $H = AX$, where A is an $m \times r$ unknown array response of full rank $r \leq \min(m,l)$ and X is an $r \times l$ matrix of unknown basis function coefficients. The vector of the unknown parameters for this model is $\boldsymbol{\rho}_{\mathrm{u}} = [\mathrm{vec}\,(\mathrm{Re}\{H\})^T, \mathrm{vec}\,(\mathrm{Im}\{H\})^T, \boldsymbol{\eta}^T]^T$. The mean signal vector for the unstructured model at time t is then $\boldsymbol{\mu}_{\mathrm{u}}(t,\boldsymbol{\rho}_{\mathrm{u}}) = H\boldsymbol{\phi}(t,\boldsymbol{\eta})$.

3 ML Estimation with Structured Array

In this section we present ML estimation for the structured array response model in (2.2). If $\boldsymbol{\theta}$ and $\boldsymbol{\eta}$ are known, the above model is known as the generalized multivariate analysis of variance (GMANOVA) [11] and has been applied in radar array processing, see [10]. The ML method for known $\boldsymbol{\eta}$ and unknown $\boldsymbol{\theta}$ is derived in [3] and used for DOA estimation.

Define $Y = [\boldsymbol{y}(1)\cdots\boldsymbol{y}(N)]$ and $B(\boldsymbol{\eta}) = [\boldsymbol{\phi}(1,\boldsymbol{\eta})\cdots\boldsymbol{\phi}(N,\boldsymbol{\eta})]$. The ML estimates of X and Σ for known $\boldsymbol{\theta}$ and $\boldsymbol{\eta}$ are [10], [11, Chapter 6.4]

$$\begin{aligned}
\widehat{X} &= [A(\boldsymbol{\theta})^*\widehat{S}_{y|\phi}^{-1}A(\boldsymbol{\theta})]^{-1}A(\boldsymbol{\theta})^*\widehat{S}_{y|\phi}^{-1}\widehat{R}_{y\phi}\widehat{R}_{\phi\phi}^{-1}, && \text{(3.1a)}\\
\widehat{\Sigma}(\boldsymbol{\theta},\boldsymbol{\eta}) &= \widehat{S}_{y|\phi} + (I_m - \widehat{T}_A\widehat{S}_{y|\phi}^{-1})\widehat{R}_{y\phi}\widehat{R}_{\phi\phi}^{-1}\widehat{R}_{y\phi}^*(I_m - \widehat{T}_A\widehat{S}_{y|\phi}^{-1})^*, && \text{(3.1b)}
\end{aligned}$$

where $\widehat{S}_{y|\phi} = \widehat{R}_{yy} - \widehat{R}_{y\phi}\widehat{R}_{\phi\phi}^{-1}\widehat{R}_{y\phi}^*$, $\widehat{R}_{yy} = \frac{1}{N}YY^*$, $\widehat{R}_{\phi\phi} = \frac{1}{N}B(\boldsymbol{\eta})B(\boldsymbol{\eta})^*$, $\widehat{R}_{y\phi} = \widehat{R}_{\phi y}^* = \frac{1}{N}YB(\boldsymbol{\eta})^*$, $\widehat{T}_A = A(\boldsymbol{\theta})[A(\boldsymbol{\theta})^*\widehat{S}_{y|\phi}^{-1}A(\boldsymbol{\theta})]^{-1}A(\boldsymbol{\theta})^*$, and I_m denotes the identity matrix of size m. Note that $\widehat{R}_{\phi\phi}$ and $\widehat{R}_{y\phi}$ are functions of $\boldsymbol{\eta}$ only, and $\widehat{T}_A$ and $\widehat{X}$ are functions of both $\boldsymbol{\theta}$ and $\boldsymbol{\eta}$. To simplify the notation, we omit these dependencies throughout this paper. Further, observe that $\widehat{R}_{y\phi}\widehat{R}_{\phi\phi}^{-1}$ is an adaptive (i.e. estimated) Wiener filter for estimating $\boldsymbol{y}(t)$ from $\boldsymbol{\phi}(t)$, $t = 1,\ldots,N$, and $\widehat{S}_{y|\phi}$ is the estimate of its error covariance, see [12, Section VII]. Also, note that $\widehat{S}_{y|\phi}^{-\frac{1}{2}}\widehat{T}_A\widehat{S}_{y|\phi}^{-\frac{1}{2}}$ is the projection matrix onto the column space of $\widehat{S}_{y|\phi}^{-\frac{1}{2}}A(\boldsymbol{\theta})$. Here, $H^{\frac{1}{2}}$ denotes an Hermitian square root of an Hermitian matrix H, and $H^{-\frac{1}{2}} = (H^{\frac{1}{2}})^{-1}$; this notation will be used throughout the paper.

If $\boldsymbol{\theta}$ and $\boldsymbol{\eta}$ are not known, their ML estimates $\widehat{\boldsymbol{\theta}}$ and $\widehat{\boldsymbol{\eta}}$ are obtained by maximizing the concentrated likelihood function $l(\boldsymbol{\theta},\boldsymbol{\eta}) = |\widehat{R}_{yy}|/|\widehat{\Sigma}(\boldsymbol{\theta},\boldsymbol{\eta})|$ (see e.g. [13]), where $\widehat{\Sigma}(\boldsymbol{\theta},\boldsymbol{\eta})$ is the ML estimate of Σ for known $\boldsymbol{\theta}$ and $\boldsymbol{\eta}$, defined in (3.1b). This concentrated likelihood function is written in the form of a generalized likelihood ratio (GLR) test statistic for testing H_0: $X = 0$. To find the ML estimates of X and Σ for unknown $\boldsymbol{\theta}$ and $\boldsymbol{\eta}$, substitute $\boldsymbol{\theta}$ and $\boldsymbol{\eta}$ in (3.1a) and (3.1b) by $\widehat{\boldsymbol{\theta}}$ and $\widehat{\boldsymbol{\eta}}$.

The above concentrated likelihood function can be rewritten by repeatedly applying the matrix inversion lemma, see [10, pp. 26–28 and p. 24]

$$l(\boldsymbol{\theta},\boldsymbol{\eta}) = \frac{|B(\boldsymbol{\eta})[I_N - \frac{1}{N}Y^*WY]B(\boldsymbol{\eta})^*|}{|B(\boldsymbol{\eta})[I_N - \frac{1}{N}Y^*\widehat{R}_{yy}^{-1}Y]B(\boldsymbol{\eta})^*|} = \frac{|\widehat{R}_{\phi\phi} - \widehat{R}_{y\phi}^*W\widehat{R}_{y\phi}|}{|\widehat{S}_{\phi|y}|} = \frac{|A(\boldsymbol{\theta})^*\widehat{S}_{y|\phi}^{-1}A(\boldsymbol{\theta})|}{|A(\boldsymbol{\theta})^*\widehat{R}_{yy}^{-1}A(\boldsymbol{\theta})|}, \tag{3.2}$$

where $W = \widehat{R}_{yy}^{-1} - \widehat{R}_{yy}^{-1}A(\boldsymbol{\theta})[A(\boldsymbol{\theta})^*\widehat{R}_{yy}^{-1}A(\boldsymbol{\theta})]^{-1}A(\boldsymbol{\theta})^*\widehat{R}_{yy}^{-1}$ and $\widehat{S}_{\phi|y} = \widehat{R}_{\phi\phi} - \widehat{R}_{y\phi}^*\widehat{R}_{yy}^{-1}\widehat{R}_{y\phi}$. Here $S_{\phi|y}$ is a function of $\boldsymbol{\eta}$ and W is a function of $\boldsymbol{\theta}$. However we omit these dependencies throughout the paper to simplify the notation. Note that $\widehat{R}_{yy}^{\frac{1}{2}}W\widehat{R}_{yy}^{\frac{1}{2}}$ is the projection matrix onto the space orthogonal to the column space of $\widehat{R}_{yy}^{-\frac{1}{2}}A(\boldsymbol{\theta})$. When there is only one path, i.e. $P = 1$ and $K = 0$, the concentrated likelihood is obtained by replacing $A(\boldsymbol{\theta})$, X, and $\boldsymbol{\phi}(t, \boldsymbol{\eta})$ with $\boldsymbol{a}(\boldsymbol{\theta})$, x, and $\exp(j\omega_{\mathrm{D}}t)s(t-\tau)$ in equation (3.2), see [8]. For $s(t) \equiv 1$ the concentrated likelihood further reduces to the expression in [7, equation (16)].

In the following, we present conditions for equivalence between the discrete-time (see above) and continuous-time signal processing used in radar [14] and communications [1], [2]. Continuous-time results are often easier to interpret, at the cost of neglecting the finite sampling effects. They also allow for time-frequency interpretations of the ML estimation results (see below) and accuracy measures (see Section 5).

Assumption A. The received signal ($\boldsymbol{\mu}(t, \boldsymbol{\eta})$ for the structured or $\boldsymbol{\mu}_{\mathrm{u}}(t, \boldsymbol{\eta}_{\mathrm{u}})$ for unstructured array model) is completely covered by the observations $t = 1, 2, \ldots, N$ and the sampling is dense (i.e. $N \to \infty$).

From the definition of $\widehat{R}_{y\phi}$ it follows that $\widehat{R}_{y\phi} = [\widehat{\boldsymbol{r}}_{y\phi}(1) \cdots \widehat{\boldsymbol{r}}_{y\phi}(l)] = \frac{1}{N}\sum_{t=1}^{N} \boldsymbol{y}(t)\boldsymbol{\phi}(t, \boldsymbol{\eta})^* \approx \int \boldsymbol{y}'(t')\boldsymbol{\phi}'(t', \boldsymbol{\eta})^* dt'$, where the integral approximation holds under Assumption A and $t' = t/N$, $\boldsymbol{y}'(t') = \boldsymbol{y}(t'N) = \boldsymbol{y}(t)$, $\boldsymbol{\phi}'(t', \boldsymbol{\eta}) = \boldsymbol{\phi}(t'N, \boldsymbol{\eta}) = \boldsymbol{\phi}(t, \boldsymbol{\eta})$, implying that the time duration of the observation interval is fixed and equals $N' = N/N = 1$. Further, we have $\widehat{\boldsymbol{r}}_{y\phi}(c(p,k)) \approx \int \boldsymbol{y}'(t')s'(t' - \tau' - p\Delta\tau')^* \exp[-j(\omega_{\mathrm{D}}' + k\Delta\omega_{\mathrm{D}}')t']dt'$, where $c(p,k) = k + K + 1 + (2K+1)\cdot p$, $s'(t') = s(t'N)$, $\tau' = \tau/N$, $\Delta\tau' = \Delta\tau'/N$, $\omega_{\mathrm{D}}' = \omega_{\mathrm{D}}N$, and $\Delta\omega_{\mathrm{D}}' = \Delta\omega_{\mathrm{D}}N$. Therefore, under Assumption A, the columns of $\widehat{R}_{y\phi}$ are the samples of the *short-time Fourier transform* (STFT) [15, p. 94] of $\boldsymbol{y}'(t')$ for various discretized time delays and Doppler shifts with sampling intervals $\Delta\tau'$ and $\Delta\omega_{\mathrm{D}}'$, respectively. In [1], [2] the elements of $\widehat{R}_{y\phi}$ are called the canonical multipath-Doppler coordinates and can be efficiently computed by using a bank of matched filters, see [1, Figure 5]. Also, the $(c(p_1, k_1), c(p_2, k_2))$ entry of $\widehat{R}_{\phi\phi}$ is of the following form:

$$\left[\widehat{R}_{\phi\phi}\right]_{c(p_1,k_1),c(p_2,k_2)} \approx \int s'(t' - \tau' - p_1\Delta t')s'(t' - \tau' - p_2\Delta t')^* \exp[-j(k_2 - k_1)\Delta\omega_{\mathrm{D}}'t']dt'. \quad (3.3)$$

Note that the squared absolute values of the above integral expressions (for various p_1, k_1, p_2, and k_2) are the samples of the Woodward's ambiguity function for signal $s'(t')$ [14, p. 279], discretized with sampling intervals $\Delta\tau'$ and $\Delta\omega_{\mathrm{D}}'$ on the time and frequency axis, respectively. After a change of variables $r' = t' - \tau'$ the expression (3.3) reduces to $\int s'(r' - p_1\Delta t')s'(r' - p_2\Delta t')^* \exp[-j(k_2 - k_1)\Delta\omega_{\mathrm{D}}'r']dr' \cdot \exp[-j(k_2 - k_1)\Delta\omega_{\mathrm{D}}'\tau']$. The first (integral) term in this expression can be computed off-line. Therefore, its on-line computation requires only a multiplication with $\exp[-j(k_2 - k_1)\Delta\omega_{\mathrm{D}}'\tau']$ to adjust for a particular global delay τ'.

3.1 Suppression of Spatial and Temporal Interference

In this section we show how known spatial or temporal interference subspace (or both) can be easily eliminated by suitably filtering the data and modifying the array response and basis function matrices.

First, consider the case of known spatial interference, i.e. where the array response is $A(\boldsymbol{\theta}) = [A_{\mathrm{i}}, A_{\mathrm{s}}(\boldsymbol{\theta})]$ and A_{i} is the array response of a known interference. Here, we introduce the following notation for the concentrated likelihood in (3.2): $l(\boldsymbol{\theta}, \boldsymbol{\eta} \,|\, Y, A(\boldsymbol{\theta}), B(\boldsymbol{\eta}))$, meaning that it is computed for the data set Y, array response $A(\boldsymbol{\theta})$, and basis function matrix $B(\boldsymbol{\eta})$. We show in [16] that the concentrated likelihood expression in (3.2) can be decomposed into a product of the concentrated likelihood function for the data set Y and interference array response A_{i} only, and the concentrated likelihood function for the filtered data set

$\widetilde{Y}_{A_\mathrm{i}} = \Psi_{A_\mathrm{i}}^* Y$ and array response $\widetilde{A}_{A_\mathrm{i}}(\boldsymbol{\theta}) = \Psi_{A_\mathrm{i}}^* A_\mathrm{s}(\boldsymbol{\theta})$, where Ψ_{A_i} is an arbitrary full-rank matrix such that its columns span the space orthogonal to the column space of A_i, i.e. $\Psi_{A_\mathrm{i}}^* A_\mathrm{i} = 0$ (For example, Ψ_{A_i} can be chosen to be the matrix whose columns are the $m - \mathrm{rank}(A_\mathrm{i})$ eigenvectors of $\Pi_{A_\mathrm{i}}^\perp = I_m - A_\mathrm{i}[A_\mathrm{i}^* A_\mathrm{i}]^{-1} A_\mathrm{i}^* = \Psi_{A_\mathrm{i}} \Psi_{A_\mathrm{i}}^*$ with corresponding eigenvalues equal to 1). In other words, $l(\boldsymbol{\theta}, \boldsymbol{\eta} \,|\, Y, A(\boldsymbol{\theta}), B(\boldsymbol{\eta})) = l(\boldsymbol{\eta} \,|\, Y, A_\mathrm{i}, B(\boldsymbol{\eta})) \cdot l(\boldsymbol{\theta}, \boldsymbol{\eta} \,|\, \widetilde{Y}_{A_\mathrm{i}}, \widetilde{A}_{A_\mathrm{i}}(\boldsymbol{\theta}), B(\boldsymbol{\eta}))$. Note that the first term in the above product does not depend on the DOA parameters $\boldsymbol{\theta}$. Thus, to find the ML estimate of $\boldsymbol{\theta}$ we need to maximize the second term only.

Now consider the case of known temporal interference, i.e. where $B(\boldsymbol{\eta}) = [B_\mathrm{i}^T, B_\mathrm{s}(\boldsymbol{\eta})^T]^T$ and B_i is the basis function matrix of a known interference. Let Φ_i be the matrix whose columns are the $N - \mathrm{rank}(B_\mathrm{i})$ eigenvectors of $\Pi_{B_\mathrm{i}^*}^\perp = I_N - B_\mathrm{i}^* (B_\mathrm{i} B_\mathrm{i}^*)^{-1} B_\mathrm{i}$ with corresponding eigenvalues equal to 1; then $\Pi_{B_\mathrm{i}^*}^\perp = \Phi_\mathrm{i} \Phi_\mathrm{i}^*$. In [16] we show that the concentrated likelihood in (3.2) is a product of the concentrated likelihood function for the data matrix Y and interference basis function matrix B_i only, and the concentrated likelihood function for the filtered data set $\widetilde{Y}_{B_\mathrm{i}} = Y \Phi_\mathrm{i}$ and filtered basis functions matrix $\widetilde{B}_{B_\mathrm{i}}(\boldsymbol{\eta}) = B_\mathrm{s}(\boldsymbol{\eta}) \Phi_\mathrm{i}$. Using the above notation, this result can be stated as $l(\boldsymbol{\theta}, \boldsymbol{\eta} \,|\, Y, A(\boldsymbol{\theta}), B(\boldsymbol{\eta})) = l(\boldsymbol{\theta} \,|\, Y, A(\boldsymbol{\theta}), B_\mathrm{i}) \cdot l(\boldsymbol{\theta}, \boldsymbol{\eta} \,|\, \widetilde{Y}_{B_\mathrm{i}}, A(\boldsymbol{\theta}), \widetilde{B}_{B_\mathrm{i}}(\boldsymbol{\eta}))$. The first term in the above product does not depend on $\boldsymbol{\eta}$. Thus, to find the ML estimate of $\boldsymbol{\eta}$ we need to maximize the second term only.

4 ML Estimation with Unstructured Array

In this section we present the ML estimates of the channel and noise covariance matrices for the unstructured array model in (2.4) for known $\boldsymbol{\eta}$, as well as the concentrated likelihood to be maximized when $\boldsymbol{\eta}$ is not known. For known $\boldsymbol{\eta}$ and real signals, these results have been derived in [9] in the context of reduced-rank multivariate linear regression. Define $P = \widehat{R}_{y\phi} \widehat{R}_{\phi\phi}^{-\frac{1}{2}}$ and $\widehat{C}_{y\phi} = \widehat{R}_{yy}^{-\frac{1}{2}} P = \widehat{R}_{yy}^{-\frac{1}{2}} \widehat{R}_{y\phi} \widehat{R}_{\phi\phi}^{-\frac{1}{2}}$, which is the estimated cross-correlation between the vectors $\widehat{R}_{yy}^{-\frac{1}{2}} \boldsymbol{y}(t)$ and $\widehat{R}_{\phi\phi}^{-\frac{1}{2}} \boldsymbol{\phi}(t, \boldsymbol{\eta})$ or the estimated *coherence matrix* between $\boldsymbol{y}(t)$ and $\boldsymbol{\phi}(t, \boldsymbol{\eta})$, see [12, Section VII]. For simplicilty of notation we omit the dependence of P and $\widehat{C}_{y\phi}$ on $\boldsymbol{\eta}$. Consider the singular value decomposition (SVD) of $\widehat{C}_{y\phi}$: $\widehat{C}_{y\phi} = \widehat{U} \widehat{\Lambda} \widehat{V}^*$, with $\widehat{U}^* \widehat{U} = \widehat{U} \widehat{U}^* = I_m$, $\widehat{V}^* \widehat{V} = \widehat{V} \widehat{V}^* = I_l$, $\widehat{\Lambda} = [\widehat{\Lambda}(m), 0]$ for $m < l$ and $\widehat{\Lambda} = [\widehat{\Lambda}(l), 0]^T$ for $m > l$, and $\widehat{\Lambda}(m) = \mathrm{diag}\{\widehat{\lambda}(1), \widehat{\lambda}(2), \ldots, \widehat{\lambda}(m)\}$, where $\widehat{\lambda}(1) \geq \widehat{\lambda}(2) \geq \ldots \geq \widehat{\lambda}(\min(m, l)) \geq 0$. Again, for notational simplicity we omit the dependence of the above quantities on $\boldsymbol{\eta}$. It can be shown that $1/(1 - \widehat{\lambda}^2(i))$, $i = 1, 2, \ldots, \min(m, l)$ are the generalized eigenvalues of $\widehat{S}_{y|\phi}^{-1}$ and $\widehat{R}_{yy}^{-1}$, sorted in non-increasing order. Then, maximizing the last expression in (3.2) with respect to unstructured array response $A(\boldsymbol{\theta}) = A$ having rank r yields

$$l_\mathrm{u}(\boldsymbol{\eta}) = \prod_{i=1}^{r} \frac{1}{1 - \widehat{\lambda}^2(i)}, \tag{4.1}$$

see also [9], [10, p. 25], [16]. We now present the ML estimate of the reduced-rank channel $H = AX$. It is not possible to find unique estimates of A and X, but their product H is unique [16]. First, we adopt the following notation: $U(r)$ is the matrix containining the first r columns of U. The ML estimates of the channel and the noise covariance, for known $\boldsymbol{\eta}$, are

$$\widehat{H}(\boldsymbol{\eta}) = \widehat{R}_{yy}^{\frac{1}{2}} \widehat{U}(r) \widehat{U}(r)^* \widehat{R}_{yy}^{-\frac{1}{2}} \widehat{R}_{y\phi} \widehat{R}_{\phi\phi}^{-1} = P \widehat{V}(r) \widehat{V}(r)^* \widehat{R}_{\phi\phi}^{-\frac{1}{2}} = \widehat{R}_{yy}^{\frac{1}{2}} \widehat{U}(r) \widehat{\Lambda}(r) \widehat{V}(r)^* \widehat{R}_{\phi\phi}^{-\frac{1}{2}}, \tag{4.2a}$$

$$\widehat{\Sigma}(\boldsymbol{\eta}) = \widehat{R}_{yy} - P \widehat{V}(r) \widehat{V}(r)^* P^* = \widehat{R}_{yy} - \widehat{R}_{yy}^{\frac{1}{2}} \widehat{U}(r) \widehat{\Lambda}^2(r) \widehat{U}(r)^* \widehat{R}_{yy}^{\frac{1}{2}}, \tag{4.2b}$$

see also [9, equations (34) and (36)]. When $r = \min(m, l)$, i.e. the channel matrix H is of full rank, its estimate is simply the estimated Wiener filter $\widehat{R}_{y\phi} \widehat{R}_{\phi\phi}^{-1}$. The reduced-rank channel estimates in equation (4.2a) can then be viewed as a product of an idempotent matrix $\widehat{R}_{yy}^{\frac{1}{2}} \widehat{U}(r) \widehat{U}(r)^* \widehat{R}_{yy}^{-\frac{1}{2}}$ and the full-rank channel estimate.

If $\boldsymbol{\eta}$ is unknown, its ML estimate $\widehat{\boldsymbol{\eta}}$ is obtained by maximizing the concentrated likelihood in (4.1). To find the ML estimates of H and Σ, replace $\boldsymbol{\eta}$ in (4.2) by $\widehat{\boldsymbol{\eta}}$.

Note also that $\widehat{\lambda}(i), i = 1, \dots, r$ are the estimated *canonical correlations*, i.e. $\widehat{\lambda}(i)$ is the cosine of the angle between the i-th components of the estimated *canonical coordinates*, defined as $\widehat{\boldsymbol{y}}_{\mathrm{c}}(t) = \widehat{U}(r)^* \widehat{R}_{yy}^{-\frac{1}{2}} \boldsymbol{y}(t)$ and $\widehat{\boldsymbol{\phi}}_{\mathrm{c}}(t) = \widehat{V}(r)^* \widehat{R}_{\phi\phi}^{-\frac{1}{2}} \boldsymbol{\phi}(t, \widehat{\boldsymbol{\eta}})$, where $t = 1, 2, \dots, N$, see [12]. Furthermore $[Y - \widehat{H}(\widehat{\boldsymbol{\eta}})B(\widehat{\boldsymbol{\eta}})] \cdot B(\widehat{\boldsymbol{\eta}})^* \widehat{H}(\widehat{\boldsymbol{\eta}})^* = 0$ can be viewed as orthogonality between fit and residuals, which is an extension of a well-known univariate result.

For negligible delay and Doppler spread, i.e. when $P = 1$ and $K = 0$, the channel response matrix reduces to a vector $\boldsymbol{h}$, and the concentrated likelihood for the unstructured array is $l_{\mathrm{u1}}(\boldsymbol{\eta}) = \widehat{\boldsymbol{r}}_{y\phi}^* \widehat{R}_{yy}^{-1} \widehat{\boldsymbol{r}}_{y\phi} / \widehat{r}_{\phi\phi}$, where $\widehat{\boldsymbol{r}}_{y\phi} = \frac{1}{N} \sum_{t=1}^{N} \boldsymbol{y}(t) s(t-\tau)^* \exp(-j\omega_{\mathrm{D}} t)$, and $\widehat{r}_{\phi\phi} = \frac{1}{N} \sum_{t=1}^{N} |s(t-\tau)|^2$, see [8], [16]. Under Assumption A, $l_{\mathrm{u1}}(\boldsymbol{\eta})$ can be viewed as a multivariate extension of the *spectrogram.* Indeed, for one sensor, it reduces to the spectrogram, i.e. squared magnitude of the STFT [15, p. 95]. Note that, since the channel vector $\boldsymbol{h}$ is unstructured, it allows for an arbitrary angular spread, although it is rather artificial to expect an angular spread when the delay spread is negligible.

5 Cramér-Rao Bound

We first present a general CRB expression for the structured array model and arbitrary basis functions. Then we specialize it to the particular choice of basis functions in equation (2.3), and further to the continuous signal model that allows for time-frequency interpretation of the obtained results.

Define the projection matrices $\Pi_A^\perp(\boldsymbol{\theta}, \Sigma) = I_m - \Sigma^{-\frac{1}{2}} A(\boldsymbol{\theta})[A(\boldsymbol{\theta})^* \Sigma^{-1} A(\boldsymbol{\theta})]^{-1} A(\boldsymbol{\theta})^* \Sigma^{-\frac{1}{2}}$ and $\Pi_{B^*}^\perp(\boldsymbol{\eta}) = I_N - B(\boldsymbol{\eta})^*[B(\boldsymbol{\eta})B(\boldsymbol{\eta})^*]^{-1} B(\boldsymbol{\eta})$ which span the spaces orthogonal to the the column spaces of $\Sigma^{-\frac{1}{2}} A(\boldsymbol{\theta})$ and $B(\boldsymbol{\eta})^*$, respectively. Also, define $D_A(\boldsymbol{\theta}) = \partial \mathrm{vec}\,(A(\boldsymbol{\theta}))/\partial \boldsymbol{\theta}^T$ and $D_B(\boldsymbol{\eta}) = \partial \mathrm{vec}\,(B(\boldsymbol{\eta}))/\partial \boldsymbol{\eta}^T$. In [16] we derive the CRB for $\boldsymbol{\theta}$ and $\boldsymbol{\eta}$ and show that they are uncoupled:

$$\mathrm{CRB}_{\theta\theta} = \frac{1}{2N}\left[\mathrm{Re}\left\{ D_A(\boldsymbol{\theta})^* \cdot \left[(X \widehat{R}_{\phi\phi} X^*)^T \otimes \Sigma^{-\frac{1}{2}} \Pi_A^\perp(\boldsymbol{\theta}, \Sigma) \Sigma^{-\frac{1}{2}} \right] \cdot D_A(\boldsymbol{\theta}) \right\} \right]^{-1}, \tag{5.1a}$$

$$\mathrm{CRB}_{\eta\theta} = 0, \tag{5.1b}$$

$$\begin{aligned}
\mathrm{CRB}_{\eta\eta} &= \frac{1}{2}\Bigg[\mathrm{Re}\Bigg\{ \sum_{t=1}^{N} \frac{\partial \boldsymbol{\phi}(t, \boldsymbol{\eta})^*}{\partial \boldsymbol{\eta}} X^* A(\boldsymbol{\theta})^* \Sigma^{-1} A(\boldsymbol{\theta}) X \frac{\partial \boldsymbol{\phi}(t, \boldsymbol{\eta})}{\partial \boldsymbol{\eta}^T} \\
&\quad - \frac{1}{N} \sum_{t_2=1}^{N} \sum_{t_1=1}^{N} \boldsymbol{\phi}(t_2, \boldsymbol{\eta})^* \widehat{R}_{\phi\phi}^{-1} \boldsymbol{\phi}(t_1, \boldsymbol{\eta}) \cdot \frac{\partial \boldsymbol{\phi}(t_1, \boldsymbol{\eta})^*}{\partial \boldsymbol{\eta}} X^* A(\boldsymbol{\theta})^* \Sigma^{-1} A(\boldsymbol{\theta}) X \frac{\partial \boldsymbol{\phi}(t_2, \boldsymbol{\eta})}{\partial \boldsymbol{\eta}^T} \Bigg\} \Bigg]^{-1} \\
&= \frac{1}{2}\left[\mathrm{Re}\left\{ D_B(\boldsymbol{\eta})^* \cdot \left[\Pi_{B^*}^\perp(\boldsymbol{\eta})^T \otimes X^* A(\boldsymbol{\theta})^* \Sigma^{-1} A(\boldsymbol{\theta}) X \right] \cdot D_B(\boldsymbol{\eta}) \right\} \right]^{-1}.
\end{aligned} \tag{5.2b}$$

As a result of CRB for $\boldsymbol{\theta}$ and $\boldsymbol{\eta}$ being block-diagonal, $\mathrm{CRB}_{\eta\eta}$ remains the same regardless of whether or not $\boldsymbol{\theta}$ is known; similarly, $\mathrm{CRB}_{\theta\theta}$ is the same regardless of whether or not $\boldsymbol{\eta}$ is known. Note that the CRB expressions in (5.1) are derived for the general model in (2.2) and are thus applicable to various basis function models. For example, they may be used to compute the CRB for the model in (2.1). For the basis functions in (2.3), we have $\partial \boldsymbol{\phi}(t, \boldsymbol{\eta})/\partial \boldsymbol{\eta}^T = [jt\boldsymbol{\phi}(t, \boldsymbol{\eta}), -\boldsymbol{\phi}_{\mathrm{d}}(t, \boldsymbol{\eta})]$ and $\boldsymbol{\phi}_{\mathrm{d}}(t, \boldsymbol{\eta})$ is of the form (2.3) where $s(t)$ is replaced with $d(t) = ds(t)/dt$. Then, equation (5.1c) reduces to

$$\mathrm{CRB}_{\eta\eta} = \frac{1}{2}\begin{bmatrix} N^3 \cdot \mathrm{tr}[\mathcal{B} \cdot X^* A(\boldsymbol{\theta})^* \Sigma^{-1} A(\boldsymbol{\theta}) X] & N \cdot \mathrm{Im}\left\{ \mathrm{tr}[\Delta \cdot X^* A(\boldsymbol{\theta})^* \Sigma^{-1} A(\boldsymbol{\theta}) X \right\} \\ N \cdot \mathrm{Im}\left\{ \mathrm{tr}[\Delta \cdot X^* A(\boldsymbol{\theta})^* \Sigma^{-1} A(\boldsymbol{\theta}) X \right\} & \frac{1}{N} \cdot \mathrm{tr}[\Xi X^* A(\boldsymbol{\theta})^* \Sigma^{-1} A(\boldsymbol{\theta}) X] \end{bmatrix}^{-1},$$

where $\mathcal{B} = F_2 - F_1 \widehat{R}_{\phi\phi}^{-1} F_1$, $\Delta = C_1 - F_1 \widehat{R}_{\phi\phi}^{-1} C$, $\Xi = L - C^* \widehat{R}_{\phi\phi}^{-1} C$, $F_1 = \frac{1}{N^2}\sum_{t=1}^{N} t\boldsymbol{\phi}(t,\boldsymbol{\eta})\boldsymbol{\phi}(t,\boldsymbol{\eta}')^* \approx \int t'\boldsymbol{\phi}'(t',\boldsymbol{\eta}')\boldsymbol{\phi}'(t',\boldsymbol{\eta}')^* dt'$, $F_2 = \frac{1}{N^3}\sum_{t=1}^{N} t^2\boldsymbol{\phi}(t,\boldsymbol{\eta})\boldsymbol{\phi}(t,\boldsymbol{\eta})^* \approx \int (t')^2\boldsymbol{\phi}'(t',\boldsymbol{\eta}')\boldsymbol{\phi}'(t',\boldsymbol{\eta}')^* dt'$, $C = \sum_{t=1}^{N} \boldsymbol{\phi}(t,\boldsymbol{\eta})\boldsymbol{\phi}_{\mathrm{d}}(t,\boldsymbol{\eta})^* \approx \int \boldsymbol{\phi}'(t',\boldsymbol{\eta}')\boldsymbol{\phi}'_{\mathrm{d}}(t',\boldsymbol{\eta})^* dt'$, $C_1 = \frac{1}{N}\sum_{t=1}^{N} t\boldsymbol{\phi}(t,\boldsymbol{\eta})\boldsymbol{\phi}_{\mathrm{d}}(t,\boldsymbol{\eta})^* \approx \int t'\boldsymbol{\phi}'(t',\boldsymbol{\eta}')\boldsymbol{\phi}'_{\mathrm{d}}(t',\boldsymbol{\eta}')^* dt'$, and $L = N\sum_{t=1}^{N} \boldsymbol{\phi}_{\mathrm{d}}(t,\boldsymbol{\eta})\boldsymbol{\phi}_{\mathrm{d}}(t,\boldsymbol{\eta})^* \approx \int \boldsymbol{\phi}'_{\mathrm{d}}(t',\boldsymbol{\eta}')\boldsymbol{\phi}'_{\mathrm{d}}(t',\boldsymbol{\eta}')^* dt'$.

5.1 CRB for Continuous Signals

We have shown in [16] that, under Assumption A, $\widehat{R}_{\phi\phi}$, $\mathcal{B}$, Δ, and Ξ are invariant to translation of the time origin, implying that the CRB expressions in (5.1) do not depend on the time delay τ. Also, if the noise term $\boldsymbol{e}'(t') = \boldsymbol{e}(t'N)$ is zero-mean Gaussian with constant power spectral density matrix $\mathcal{N}_\Sigma$ in the frequency band $(-N/2, N/2)$ and zero outside, then the noise covariance is $\Sigma = \Sigma(N) = \mathcal{N}_\Sigma \cdot N$ and the CRBs for $\boldsymbol{\theta}$ and $\boldsymbol{\eta}' = [\omega'_{\mathrm{D}}, \tau']^T = [\omega_{\mathrm{D}}N, \tau/N]^T$ easily follow from (5.1a) and (5.1c). The obtained expressions do not depend on N, since, under the above noise model, the noise level increases linearly with N, which cancels out the linear decrease of the CRB due to oversampling.

For negligible delay and Doppler spread, i.e. when $P = 1$ and $K = 0$, the channel response is $\boldsymbol{h} = x\boldsymbol{a}(\boldsymbol{\theta})$ for the structured or $\boldsymbol{h}$ for unstructured array and $\mathrm{CRB}_{\eta'\eta'}$ reduces to, see [16], [8]

$$\mathrm{CRB}_{\eta'\eta'} = \frac{1}{2 \cdot \boldsymbol{h}^* \mathcal{N}_\Sigma{}^{-1} \boldsymbol{h}} \cdot \begin{bmatrix} \beta & \mathrm{Im}\{\delta\} \\ \mathrm{Im}\{\delta\} & \xi \end{bmatrix}^{-1}, \tag{5.3}$$

where $\beta = \int (t')^2 |s'(t'-\tau')|^2 dt' - \frac{1}{\widehat{r}_{\phi\phi}} \cdot [\int t' |s'(t'-\tau')|^2 dt']^2$, $\delta = \int t' s'(t'-\tau') \frac{ds'(t'-\tau')^*}{dt'} dt' - \frac{1}{\widehat{r}_{\phi\phi}} \cdot \int t' |s'(t'-\tau')|^2 dt' \cdot \int s'(t'-\tau') \frac{ds'(t'-\tau')^*}{dt'} dt'$, $\xi = \int \left| \frac{ds'(t'-\tau')}{dt'} \right|^2 dt' - \frac{1}{\widehat{r}_{\phi\phi}} \left| \int s'(t'-\tau') \frac{ds'(t'-\tau')^*}{dt'} dt' \right|^2$, and $\widehat{r}_{\phi\phi} = \int |s'(t'-\tau')|^2 dt'$. Here $\widehat{r}_{\phi\phi}$ is the signal energy, and $\beta/\widehat{r}_{\phi\phi}$ and $\xi/\widehat{r}_{\phi\phi}$ are the effective duration and effective bandwidth of $s(t)$, respectively [16], [8] (see also [15, Sections 1.2 and 1.3] where the signal energy is normalized, i.e. $\widehat{r}_{\phi\phi} = 1$). Also, $-\mathrm{Im}\{\delta\}/\widehat{r}_{\phi\phi}$ can be viewed as the covariance of $s'(t')$ [16], [8], i.e. the measure of how time is correlated with instantaneous frequency. Indeed, for normalized signal energy, it reduces to the signal covariance as defined in [15, equation (1.124)].

6 Receivers

Recently, several authors have incorporated the spatial noise covariance into the receiver design. In [4] and [5] the maximum likelihood sequence estimators (MLSE) which account for spatially correlated interference are derived. Here, we propose coherent matched-filter and decision-feedback receivers that utilize channel and spatial noise covariance estimates from Sections 3 and 4. We also discuss the relationship between the two receivers.

Define the data sets Y_{K} (of size $m \times N_{\mathrm{K}}$) and Y_{R} (of size $m \times N_{\mathrm{R}}$) containing the data with known training (or previously detected) sequence, and the unknown symbol to be detected, respectively. Then, for J symbols, denote the basis function matrices of the symbol to be detected as B_j, where $j = 0, 1, 2, \ldots, J-1$. In the discussion below, we assume that synchronization to the global time delay and Doppler shift has been done, i.e. $\boldsymbol{\eta}$ is known.

6.1 Coherent Matched-filter Receiver

We present a coherent matched-filter receiver which accounts for spatially correlated interference. The ML methods in Sections 3 and 4 are applied to the training data Y_{K} to estimate the channel and spatial covariance matrix, denoted by $\widehat{H}_{\mathrm{K}}$ and $\widehat{\Sigma}_{\mathrm{K}}$, respectively. Then, the coherent matched-filter receiver is based on choosing the j that maximizes the following decision statistic: $m_j = 2\,\mathrm{Re}\{\,\mathrm{tr}[Y_{\mathrm{R}}^{*}\widehat{\Sigma}_{\mathrm{K}}^{-1}\widehat{H}_{\mathrm{K}}B_j]\} - \mathrm{tr}[B_j{}^{*}\widehat{H}_{\mathrm{K}}^{*}\widehat{\Sigma}_{\mathrm{K}}^{-1}\widehat{H}_{\mathrm{K}}B_j]$, $j = 0, 1, \ldots, J-1$. For spatially uncorrelated noise (i.e. setting $\widehat{\Sigma}_{\mathrm{K}} = \sigma^2 I$ in the above expression for m_j), the above recevier reduces to the coherent matched-filter receiver in [2]. For negligible Doppler spread (i.e. $K = 0$) and the model in (2.1), it further reduces to the space-time RAKE receiver [17, equation (111)]. If the channel and spatial noise covariance parameters are known exactly, we show in [16] that the above receiver performs better (or equal to) the receiver for spatially uncorrelated noise, in terms of error probability.

For the binary case (i.e. $J = 2$), the decision is made by computing $\mathrm{sign}(m_1 - m_0) = \mathrm{sign}(\,\mathrm{Re}\{\,\mathrm{tr}\,[2Y_{\mathrm{R}}^{*}\widehat{\Sigma}_{\mathrm{K}}^{-1}\widehat{H}_{\mathrm{K}}(B_1 - B_0) - (B_1 + B_0)^{*}\widehat{H}_{\mathrm{K}}^{*}\widehat{\Sigma}_{\mathrm{K}}^{-1}\widehat{H}_{\mathrm{K}}(B_1 - B_0)]\})$. The second term in this expression disappears if antipodal signaling is employed (i.e. $B_1 = -B_0$), or 1 and 0 bases have the same covariance matrices (i.e. $\widehat{R}_{\phi_0\phi_0} = \frac{1}{N}B_0B_0{}^{*} = \widehat{R}_{\phi_1\phi_1} = \frac{1}{N}B_1B_1{}^{*}$). Then, the detector is simply $\mathrm{sign}(\,\mathrm{Re}\{\,\mathrm{tr}\,[Y_{\mathrm{R}}^{*}\widehat{\Sigma}_{\mathrm{K}}^{-1}\widehat{H}_{\mathrm{K}}(B_1 - B_0)]\})$.

If $\widehat{H}_{\mathrm{K}}$ and $\widehat{\Sigma}_{\mathrm{K}}$ are the unstructured channel and noise estimates from equation (4.2) applied to Y_{K}, then $\mathrm{tr}[Y_{\mathrm{R}}^{*}\widehat{\Sigma}_{\mathrm{K}}^{-1}\widehat{H}_{\mathrm{K}}B_j] = \sum_{t=1}^{N}\widehat{\boldsymbol{y}}_{R\mathrm{c}}(t)^{*}[I_r - \widehat{\Lambda}_{\mathrm{K}}^2(r)]^{-1}\widehat{\Lambda}_{\mathrm{K}}(r)\widehat{\boldsymbol{\phi}}_{j\mathrm{c}}(t) = \sum_{t=1}^{N}\sum_{i=1}^{r}\widehat{y}_{R\mathrm{c},i}(t)^{*}\cdot$ $\cdot\widehat{\lambda}_{\mathrm{K}}(i)/[1 - \widehat{\lambda}_{\mathrm{K}}^2(i)]\cdot\widehat{\phi}_{j\mathrm{c},i}(t)$ and $\mathrm{tr}[B_j{}^{*}\widehat{H}_{\mathrm{K}}^{*}\widehat{\Sigma}_{\mathrm{K}}^{-1}\widehat{H}_{\mathrm{K}}B_j] = \sum_{t=1}^{N}\sum_{i=1}^{r}\widehat{\phi}_{j\mathrm{c},i}(t)^{*}\cdot\widehat{\lambda}_{\mathrm{K}}^2(i)/[1 - \widehat{\lambda}_{\mathrm{K}}^2(i)]\cdot$ $\cdot\widehat{\phi}_{j\mathrm{c},i}(t)$, where $\widehat{\lambda}_{\mathrm{K}}(i)$ are the canonical correlations, and $\widehat{\boldsymbol{y}}_{R\mathrm{c}}(t)$ and $\widehat{\boldsymbol{\phi}}_{j\mathrm{c}}(t)$ are the canonical coordinates, estimated from the training data Y_{K}. Using the results of Section 4, these quantities are easily computed as $\widehat{\boldsymbol{y}}_{\mathrm{Rc}}(t) = \widehat{U}_{\mathrm{K}}(r)^{*}\widehat{R}_{\mathrm{K}yy}^{-\frac{1}{2}}\boldsymbol{y}_{\mathrm{R}}(t)$, and $\widehat{\boldsymbol{\phi}}_{j\mathrm{c}}(t) = \widehat{V}_{\mathrm{K}}(r)^{*}\widehat{R}_{\mathrm{K}\phi\phi}^{-\frac{1}{2}}\boldsymbol{\phi}_j(t)$, where $\widehat{C}_{\mathrm{K}y\phi} = \widehat{R}_{\mathrm{K}yy}^{-\frac{1}{2}}\widehat{R}_{\mathrm{K}y\phi}\widehat{R}_{\mathrm{K}\phi\phi}^{-\frac{1}{2}} = \widehat{U}_{\mathrm{K}}\widehat{\Lambda}_{\mathrm{K}}\widehat{V}_{\mathrm{K}}^{*}$, $\widehat{R}_{\mathrm{K}yy} = Y_{\mathrm{K}}Y_{\mathrm{K}}{}^{*}/N_{\mathrm{K}}$, $\widehat{R}_{\mathrm{K}\phi\phi} = B_{\mathrm{K}}B_{\mathrm{K}}{}^{*}/N_{\mathrm{K}}$, $\widehat{R}_{\mathrm{K}y\phi} = Y_{\mathrm{K}}B_{\mathrm{K}}{}^{*}/N_{\mathrm{K}}$. Thus, for the unstructured channel, the coherent matched-filter reception can be viewed as matching the estimated canonical coordinates, suitably weighted by the corresponding canonical correlations.

6.2 Decision-Feedback Receiver

Here we propose an adaptive receiver structure which utilizes several previously detected symbols and the currently received symbol to estimate the noise and channel parameters. Then, the likelihood function concentrated with respect to these parameters is used to detect the new symbol. For simplicity, we assume a full-rank unstructured channel (see Section 4) which allows for an easy recursive computation of the concentrated likelihood, as shown in Section 6.2.2. Define $Y = [Y_{\mathrm{K}}, Y_{\mathrm{R}}]$ containing $N = N_{\mathrm{K}} + N_{\mathrm{R}}$ snapshots, and the corresponding basis function matrix for the received symbol j as $B = [B_{\mathrm{K}}, B_j]$, where $j \in \{0, 1, 2, \ldots, J-1\}$. Note that $B_j = [B_{\mathrm{K}j}, B_{0j}]$, where $B_{\mathrm{K}j}$ is the basis function matrix covering the part of the time where there is an overlap between the known and unknown bases due to multipath spread, and B_{0j} is the basis function matrix for the unknown symbol only. In CDMA applications, $B_{\mathrm{K}j}$ usually covers only a small part of the symbol duration.

Under the above assumptions, the two forms of concentrated likelihood function easily follow from (3.2) as

$$l = \frac{|\widehat{R}_{\phi\phi}|}{|\widehat{S}_{\phi|y}|} = \frac{|\widehat{R}_{\phi\phi}|}{|\widehat{R}_{\phi\phi} - \widehat{R}_{y\phi}^{*}\widehat{R}_{yy}^{-1}\widehat{R}_{y\phi}|} = \frac{|\widehat{R}_{yy}|}{|\widehat{S}_{y|\phi}|} = \frac{|\widehat{R}_{yy}|}{|\widehat{R}_{yy} - \widehat{R}_{y\phi}\widehat{R}_{\phi\phi}^{-1}\widehat{R}_{y\phi}^{*}|}, \tag{6.1}$$

where $\widehat{R}_{\phi\phi} = \frac{1}{N}BB^{*} = \frac{1}{N}(B_{\mathrm{K}}B_{\mathrm{K}}{}^{*} + B_jB_j{}^{*})$, $\widehat{R}_{yy} = \frac{1}{N}YY^{*} = \frac{1}{N}(Y_{\mathrm{K}}Y_{\mathrm{K}}{}^{*} + Y_{\mathrm{R}}Y_{\mathrm{R}}{}^{*})$, and $\widehat{R}_{y\phi} = \frac{1}{N}YB^{*} = \frac{1}{N}(Y_{\mathrm{K}}B_{\mathrm{K}}{}^{*} + Y_{\mathrm{R}}B_j{}^{*})$. The receiver detects the new symbol by choosing the j that maximizes the concentrated likelihood function in (6.1) with respect to $j = 0, 1, 2, \ldots, J-1$.

We have shown in [16] that the coherent matched-filter receiver for an unstructured channel of full rank is an approximation of the above decision-feedback receiver assuming $N_{\mathrm{K}} >> N_{\mathrm{R}}$, i.e. that the known symbol sequence is long. In particular, we have shown that, under this assumption, $\ln l - \ln|N\widehat{R}_{yy}| + \ln|N_{\mathrm{K}}\widehat{S}_{\mathrm{K}y|\phi}| \approx -\frac{1}{N_{\mathrm{K}}}\,\mathrm{tr}\,\{\widehat{S}_{\mathrm{K}y|\phi}^{-1}[Y_{\mathrm{R}} -$

$\widehat{H}_{\mathrm{K}}B_j] \cdot [Y_{\mathrm{R}} - \widehat{H}_{\mathrm{K}}B_j]^*\}$, where l is defined in (6.1), and $\widehat{H}_{\mathrm{K}} = Y_{\mathrm{K}}B_{\mathrm{K}}{}^*(B_{\mathrm{K}}B_{\mathrm{K}}{}^*)^{-1}$ and $\widehat{S}_{\mathrm{K}y|\phi} = \widehat{\Sigma}_{\mathrm{K}} = \frac{1}{N_{\mathrm{K}}}[Y_{\mathrm{K}}Y_{\mathrm{K}}{}^* - Y_{\mathrm{K}}B_{\mathrm{K}}{}^*(B_{\mathrm{K}}B_{\mathrm{K}}{}^*)^{-1}B_{\mathrm{K}}Y_{\mathrm{K}}{}^*]$ are the ML estimates of the channel and noise covariance matrices for the full-rank unstructured channel. Then, the coherent matched-filter decision statistic in Section 6.1 immediately follows.

6.2.1 Channel with Negligible Delay and Doppler Spread

To gain more insight about the proposed receiver, we consider a special case with negligible delay and Doppler spread (i.e. $P = 1$ and $K = 0$). For simplicity, we also assume that Assumption A holds. Denote the known signal by $s'_{\mathrm{K}}(t')$ (for $0 \le t' \le N'_{\mathrm{K}}$) and the jth symbol waveshape by $s'_j(t' - N'_{\mathrm{K}})$, for $N'_{\mathrm{K}} \le t' \le 1$ and $j = 0, 1, 2, \ldots, J-1$. Then, $\widehat{\boldsymbol{r}}_{y\phi} = \widehat{\boldsymbol{r}}_{\mathrm{K}y\phi} + \widehat{\boldsymbol{r}}_{jy\phi} = \int_0^{N'_{\mathrm{K}}} \boldsymbol{y}'(t')s'_{\mathrm{K}}(t')^* \exp(-j\omega'_{\mathrm{D}}t')dt' + \int_{N'_{\mathrm{K}}}^1 \boldsymbol{y}'(t')s'_j(t' - N'_{\mathrm{K}})^* \exp(-j\omega'_{\mathrm{D}}t')dt'$, $\widehat{R}_{yy} = \widehat{R}_{\mathrm{K}yy} + \widehat{R}_{\mathrm{R}yy} = \int_{t'=0}^{N'_{\mathrm{K}}} \boldsymbol{y}'(t')\boldsymbol{y}'(t')^* + \int_{t'=N'_{\mathrm{K}}}^1 \boldsymbol{y}'(t')\boldsymbol{y}'(t')^* = \int_{t'=0}^1 \boldsymbol{y}'(t')\boldsymbol{y}'(t')^*$, and the decision statistic easily follows from $l_{\mathrm{u1}}(\boldsymbol{\eta})$ in Section 4 as $m_j = 2\,\mathrm{Re}\,\{\widehat{\boldsymbol{r}}^*_{\mathrm{K}y\phi}\widehat{R}^{-1}_{yy}\widehat{\boldsymbol{r}}_{jy\phi}\} + \widehat{\boldsymbol{r}}^*_{jy\phi}\widehat{R}^{-1}_{yy}\widehat{\boldsymbol{r}}_{jy\phi}$, where $j = 0, 1, \ldots, J-1$. Note that $\widehat{\boldsymbol{r}}^*_{\mathrm{K}y\phi}\widehat{R}^{-1}_{yy}$ above resembles the celebrated sample matrix inversion (SMI) beamformer [18] for the reference signal $s'_{\mathrm{K}}(t')$. It would be equal to the SMI beamformer if $\widehat{R}_{yy}$ is replaced by the sample covariance $\widehat{R}_{\mathrm{K}yy}$, estimated from the training data only. In the case of binary antipodal signaling, i.e. $J = 2$ and $s'_1(t) = -s'_0(t')$, the above detector further reduces to $\mathrm{sign}(\,\mathrm{Re}\,\{\widehat{\boldsymbol{r}}^*_{\mathrm{K}y\phi}\widehat{R}^{-1}_{yy}\widehat{\boldsymbol{r}}_{1y\phi}\})$.

6.2.2 Recursive Implementation

In this section we propose the recursive implementation of the decision-feedback receiver. Define the data set and basis function matrices $Y(t) = [Y(t-1), \boldsymbol{y}(t)]$ and $B_t = [B(t-1), \boldsymbol{\phi}(t)]$, both containing t snapshots. We denote by l_t the concentrated likelihood in (6.1) computed for $Y = Y(t)$ and $B = B(t)$. We also define $P_{yyt} = [Y(t)Y(t)^*]^{-1}$, $P_{\phi\phi t} = [B(t)B(t)^*]^{-1}$, $Q_t = P_{yyt}Y(t)B(t)^*$, $P_{\phi|yt} = [B(t)B(t)^* - B(t)Y(t)^*Q_t]^{-1}$. Then, in [16] we derive the following recursion, based on the first form of the concentrated likelihood (6.1):

$$\gamma(t) = \frac{1}{1 + \boldsymbol{y}(t)^*P_{yy(t-1)}\boldsymbol{y}(t)}, \tag{6.2a}$$

$$l_t = l_{t-1} \cdot \frac{1 + \boldsymbol{\phi}(t)^*P_{\phi\phi(t-1)}\boldsymbol{\phi}(t)}{1 + \gamma(t) \cdot [\boldsymbol{\phi}(t) - Q_{t-1}^*\boldsymbol{y}(t)]^*P_{\phi|y(t-1)}[\boldsymbol{\phi}(t) - Q_{t-1}^*\boldsymbol{y}(t)]}. \tag{6.2b}$$

Updates of the following quantities prepare for the next step:

$$P_{yyt} = P_{yy(t-1)} - \frac{P_{yy(t-1)}\boldsymbol{y}(t)\boldsymbol{y}(t)^*P_{yy(t-1)}}{1 + \boldsymbol{y}(t)^*P_{yy(t-1)}\boldsymbol{y}(t)}, \tag{6.3a}$$

$$Q_t = Q_{t-1} + P_{yyt}\,\boldsymbol{y}(t) \cdot [\boldsymbol{\phi}(t)^* - \boldsymbol{y}(t)^*Q_{t-1}], \tag{6.3b}$$

$$P_{\phi\phi t} = P_{\phi\phi(t-1)} - \frac{P_{\phi\phi(t-1)}\boldsymbol{\phi}(t)\boldsymbol{\phi}(t)^*P_{\phi\phi(t-1)}}{1 + \boldsymbol{\phi}(t)^*P_{\phi\phi(t-1)}\boldsymbol{\phi}(t)}, \tag{6.3c}$$

$$P_{\phi|yt} = P_{\phi|y(t-1)} - \gamma(t)\frac{P_{\phi|y(t-1)}[\boldsymbol{\phi}(t) - Q_{t-1}^*\boldsymbol{y}(t)][\boldsymbol{\phi}(t) - Q_{t-1}^*\boldsymbol{y}(t)]^*P_{\phi|y(t-1)}}{1 + \gamma(t)[\boldsymbol{\phi}(t) - Q_{t-1}^*\boldsymbol{y}(t)]^*P_{\phi\phi(t-1)}[\boldsymbol{\phi}(t) - Q_{t-1}^*\boldsymbol{y}(t)]}. \tag{6.3d}$$

An analogous recursion can be derived to compute the concentrated likelihood function based on the second form of the concentrated likelihood in (6.1). It essentially reduces to swapping $\boldsymbol{y}(t)$ and $\boldsymbol{\phi}(t)$ in the expressions (6.2) and (6.3).

References

[1] A.M. Sayeed and B. Aazhang, "Joint multipath-Doppler diversity in mobile wireless communications," *IEEE Trans. Commun.*, vol. 47, pp. 123–132, Jan. 1999.

[2] A.M. Sayeed, E.N. Onggosanusi, and B.D. Van Veen, "A canonical space-time characterization of mobile wireless channels," *IEEE Commun. Letters*, vol. 3, pp. 94–96, Apr. 1999.

[3] M. Viberg, P. Stoica, and B. Ottersten, "Maximum likelihood array processing in spatially correlated noise fields using parameterized signals," *IEEE Trans. Signal Processing,* vol. 45, pp. 996–1004, Apr. 1997.

[4] G.E. Bottomley and K. Jamal, "Adaptive arrays and MLSE equalization," *Proc. 45th Vehicular Technology Conference,* pp. 50–54, Chicago, IL, July 1995.

[5] M. Escartin amd P.A. Ranta, " Interference rejection with a small antenna array at the mobile scattering environment," *Proc. 1st SPAWC,* pp. 165–168, Paris, April 1997.

[6] A. Jakobsson, A.L. Swindlehurst, D. Astély, and C. Tidestav, "A blind frequency domain method for DS-CDMA synchronization using antenna arrays," *Proc. 32nd Asilomar Conf. Signals, Syst. Comput.*, Pacific Grove, CA, Nov. 1998.

[7] A.L. Swindlehurst and P. Stoica, "Maximum likelihood methods in radar array signal processing," *Proc. IEEE,* vol. 86, pp. 421–441, Feb. 1998.

[8] A. Dogandžić and A. Nehorai, "Estimating range, velocity, and direction with a radar array," *Proc. Int. Conf. Acoust., Speech, Signal Processing (ICASSP99),* pp. 2773–2776, Phoenix, AZ, March 1999.

[9] P. Stoica and M. Viberg, "Maximum likelihood parameter and rank estimation in reduced-rank multivariate linear regressions," *IEEE Trans. Signal Processing,* vol. 44, pp. 3069–3078, Dec. 1996.

[10] E.J. Kelly and K.M. Forsythe, "Adaptive detection and parameter estimation for multidimensional signal models," Technical Report 848, Lincoln Laboratory, Massachusetts Institute of Technology, Lexington, MA, April 1989.

[11] M.S. Srivastava and C.G. Khatri, *An Introduction to Multivariate Statistics,* New York, NY: North Holland, 1979.

[12] L.L. Scharf and J.K. Thomas, "Wiener filters in canonical coordinates for transform coding, filtering, and quantizing," *IEEE Trans. Signal Processing,* vol. 46, pp. 647–654, March 1998.

[13] A. Dogandžić and A. Nehorai, "Estimating evoked dipole responses in unknown spatially correlated noise with EEG/MEG arrays," to appear, *IEEE Trans. Signal Processing,* 2000.

[14] H.L. Van Trees, *Detection, Estimation and Modulation Theory: Part III — Radar-Sonar Signal Processing and Gaussian Signals in Noise*, New York, NY:Wiley, 1971.

[15] L. Cohen, *Time-frequency Analysis,* Englewood Cliffs, NJ: Prentice Hall, 1995.

[16] A. Dogandžić and A. Nehorai, "Space-time fading channel estimation in unknown spatially correlated noise," in revision, *IEEE Trans. Signal Processing.*

[17] A.J. Paulraj and C.B. Papadias, "Space-time processing for wireless communications," *IEEE Signal Processing Magazine,* vol. 14, no. 6, pp. 49–83, Nov. 1997.

[18] I.S. Reed, J.D. Mallet, and L.E. Brennan, "Rapid convergence rate in adaptive arrays," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 10, pp. 853–863, 1974.

BLAST Training: Estimating Channel Characteristics for High Capacity Space-Time Wireless

THOMAS L. MARZETTA

Mathematical Sciences Research Center
Bell Laboratories, Lucent Technologies
600 Mountain Avenue
Murray Hill, NJ 07974
tlm@research.bell-labs.com

Abstract

BLAST (Bell Labs Layered Space-Time) is a multiple-antenna communication scheme whose outage capacity in a Rayleigh flat fading environment grows linearly with the minimum of the number of transmit and receive antennas, with no increase in bandwidth or transmitted power. Based on its knowledge of the matrix of propagation coefficients, the receiver performs two critical operations: nulling and cancellation, that in effect create independent virtual subchannels.

Assume that the receiver estimates the propagation matrix from a known set of transmitted training signals, and then uses the estimate as though it were correct for nulling and cancellation. How much training is needed for satisfactory operation?

The optimal training signals are orthogonal with respect to time among the transmit antennas, and each transmit antenna is fed equal energy. Errors in estimating the propagation matrix manifest themselves as crosstalk among the virtual subchannels. If its magnitude is too large, the crosstalk constitutes an outage event that is independent of the primary outage event (e.g., that the value of the propagation matrix cannot support the transmission rate). We show that the training interval required to control the probability of the estimation-error induced outage is approximately proportional to the number of transmit antennas, and is independent of the number of receive antennas. Contrary to what is implied by their names, the operations of nulling and cancellation are, in fact, robust with respect to estimation errors.

Our results have an interesting implication for BLAST, either in a mobile environment, or for TDMA, where both training and data transmission have to occur within a fixed interval: if one wishes to maximize the overall transmission rate, then the number of transmit antennas should be chosen such that half of the interval is used for training, and half of the interval for data transmission.

1 Introduction

Rayleigh flat fading can be beneficial for a multiple-antenna communication link. Specifically, Telatar [1] and Foschini [2] showed that, in a Rayleigh flat-fading environment, a link has a theoretical capacity that increases linearly with the smaller of the number of

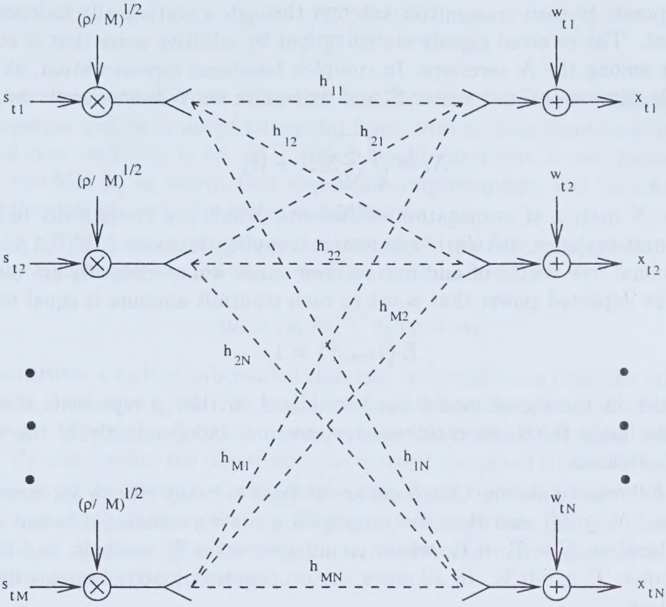


Figure 1: Wireless link comprising M transmitter and N receiver antennas. Every receiver antenna is connected to every transmitter antenna through an independent, random, propagation coefficient having Rayleigh distributed magnitude and uniformly distributed phase. Normalization ensures that the total expected transmitted power is independent of M for a fixed ρ.

transmitter and receiver antennas, provided that the complex-valued propagation coefficients between all pairs of transmitter and receiver antennas are statistically independent and known to the receiver. Moreover Foschini invented an architecture for realizing a significant fraction of this capacity, with a small probability of outage, using only ordinary modulation and coding techniques. This theory was confirmed strikingly in experimental demonstrations at Bell Labs of a particular embodiment called BLAST (Bell Labs Layered Space Time), where data was sent at a rate of 600 kbits/s, in a 30 kHz bandwidth, over a link comprising 8 transmit antennas and 12 receive antennas.

Assume that the receiver estimates the matrix of propagation coefficients from a set of known training signals that are sent by the transmitter. This paper provides answers to two related questions: 1) How much training is required for satisfactory operation of BLAST? 2) What effects do estimation errors have on the performance of the scheme?

2 Multiple-Antenna Link: Signal Model

Consider a communication link comprising M transmitter antennas and N receiver antennas that operates in a Rayleigh flat-fading environment as shown in Fig. 1. Each receiver antenna responds to each transmitter antenna through a statistically independent fading coefficient. The received signals are corrupted by additive noise that is statistically independent among the N receivers. In complex baseband representation, at time t we transmit a M-dimensional row vector S_t and we receive an N-dimensional row vector X_t, where

$$X_t = \sqrt{\frac{\rho}{M}} S_t H + W_t, \tag{1}$$

H is a $M \times N$ matrix of propagation coefficients, which are statistically independent, zero-mean, unit-variance, circularly symmetric complex Gaussian ($\mathrm{CN}(0,1)$), and W_t is a N-dimensional row vector of additive receiver noise whose elements are independent $\mathrm{CN}(0,1)$. The expected power that is fed to each transmit antenna is equal to one,

$$\mathrm{E}\left\{|s_{tm}|^2\right\} = 1. \tag{2}$$

The quantities in the signal model are normalized so that ρ represents the expected signal-to-noise ratio (SNR) at each receiver antenna, independently of the number of transmitter antennas.

In what follows, we assume that there are at least as many receive antennas as transmit antennas, $N \geq M$, and that the propagation matrix remains constant over some interval of duration $T = T_\tau + T_d$, where training occupies T_τ symbols, and data transmission occupies T_d symbols. In all cases the propagation matrix remains unknown to the transmitter.

3 Review of BLAST: Nulling and Cancellation

The message bits to be transmitted are divided equally among the M transmit antennas, and the modulation and coding for each transmit antenna occurs independently of the modulation and coding for the other transmit antennas. The message is sent at some pre-arranged rate. Thus the scheme is characterized by an outage probability, since the actual value of the propagation matrix may not support the transmission rate [5]. Assume in this section that the receiver has perfect knowledge of the propagation matrix H. The receiver performs a QR factorization of the propagation matrix, and then it implements two operations: nulling and cancellation.

The QR factorization of the propagation matrix has the form (recall that $N \geq M$):

$$H = RQ, \tag{3}$$

where R is an $M \times M$ upper triangular matrix, $r_{mn} = 0,\ m > n$, and the diagonal elements of R are real-valued and nonnegative, and Q is a $M \times N$ matrix whose row vectors are orthonormal.

The nulling operation is a coordinate rotation that multiplies the received vector by $Q^\dagger$ to produce a M-dimensional row vector Y_t that constitutes a sufficient statistic,

$$\begin{aligned} Y_t &= X_t Q^\dagger \\ &= \sqrt{\frac{\rho}{M}} S_t H Q^\dagger + W_t Q^\dagger \end{aligned}$$

$$= \sqrt{\frac{\rho}{M}} S_t R + \nu_t \tag{4}$$

where the components of ν_t are independent, CN(0, 1). The effect of nulling is to render the propagation matrix in upper triangular form, with no amplification of receiver noise. One could now multiply Y_t by R^{-1}, which would constitute a zero-forcing operation. However Foschini [2] as shown that significant improvements over zero-forcing can be obtained through an operation called cancellation.

Because of the upper-triangular structure of R, the first component of Y_t depends only on the signal that was fed to the first transmit antenna,

$$y_{t1} = (\rho/M)^{1/2} s_{t1} r_{11} + \nu_{t1}, \tag{5}$$

and it constitutes a virtual subchannel that has no interference from the other subchannels, with an SNR of $\rho r_{11}^2/M$. The sequence $\{y_{t1}\}$ is now decoded to yield the associated message bits. From the decoded message bits, the transmitted signal $\{\hat{s}_{1t}\}$ is reconstructed. By assumption the decoding error is small compared to the outage probability. Consequently, in the absence of an outage event, $\hat{s}_{t1} = s_{t1}$ with a high probability.

Inspection of (4) discloses that y_{t2} is subject to interference from the first subchannel through the off-diagonal term r_{12}, which can be removed with high probability through the following cancellation operation

$$y_{t2} - (\rho/M)^{1/2} \hat{s}_{t1} r_{12} = (\rho/M)^{1/2} s_{t2} r_{22} + \nu_{t2}. \tag{6}$$

The decoding is performed on the successive subchannels, preceeded by a cancellation of the off-diagonal terms in (4).

We represent the upper-triangular matrix R as the sum of a diagonal matrix D, and a strictly above-triangular matrix U,

$$R = D + U. \tag{7}$$

Then the cancellation operation is summarized as

$$\begin{aligned} \left(\frac{\rho}{M}\right)^{-1/2} \left[Y_t - (\rho/M)^{1/2} \hat{S}_t U\right] D^{-1} &= S_t + (\rho/M)^{-1/2} \nu_t D^{-1} + \left(S_t - \hat{S}_t\right) U D^{-1} \\ &= S_t + (\rho/M)^{-1/2} \nu_t D^{-1}, \end{aligned} \tag{8}$$

where the second equality holds if $\hat{S}_t = S_t$, i.e., if no bit-errors have occurred. Thus nulling and cancellation together produce M independent virtual subchannels, and the SNR of the m-th subchannel is equal to $\rho r_{mm}^2/M$.[1]

4 Training

We assume that $T_\tau \geq M$ symbols are used for sending known training signals, represented by a $T_\tau \times M$ matrix S, and that the receiver obtains the maximum likelihood (ML) estimate for the propagation matrix from the $T_\tau \times N$ received signal X. The ML estimate is

---

[1] Foschini has shown that it is advantageous to order the nulling and cancellation, so that at each step, the subchannel that is decoded has the greatest SNR of the remaining subchannels. We do not explicitly consider this more complicated scheme, but our subsequent analysis applies equally well to it.

$$\begin{aligned}\hat{H} &= \left(\frac{\rho}{M}\right)^{-1/2} (S^\dagger S)^{-1} S^\dagger X \\ &= H + \left(\frac{\rho}{M}\right)^{-1/2} (S^\dagger S)^{-1} S^\dagger W,\end{aligned} \tag{9}$$

where W is the matrix of receiver noise. The estimate is unbiased, the estimation error is uncorrelated among the N columns of $\hat{H}$, and the covariance of each column is

$$\begin{aligned}\mathrm{E}\left\{\tilde{h}_n \tilde{h}_n^\dagger\right\} &= \mathrm{E}\left\{\left(\hat{h}_n - h_n\right)\left(\hat{h}_n - h_n\right)^\dagger\right\} \\ &= \left(\frac{M}{\rho}\right)\left(S^\dagger S\right)^{-1}.\end{aligned} \tag{10}$$

The training signals that minimize the trace of the covariance (10) subject to the total energy constraint $\mathrm{tr}\left\{S^\dagger S\right\} = T_\tau \cdot M$, have been shown to be proportional to signals that are orthonormal among the M transmit antennas [4]. To see this we represent S according to its singular value decomposition, $S = \Phi V \Psi^\dagger$, where Φ is $T_\tau \times M$, with orthonormal columns, Ψ is $M \times M$ unitary, and V is $M \times M$ diagonal, with real nonnegative elements. The error covariance is

$$\left(\frac{M}{\rho}\right)\left(S^\dagger S\right)^{-1} = \left(\frac{M}{\rho}\right) \Psi V^{-2} \Psi^\dagger, \tag{11}$$

whose trace is proportional to $\sum_{m=1}^{M} v_{mm}^{-2}$. Subject to the equivalent energy constraint $\sum_{m=1}^{M} v_{mm}^2 = T_\tau \cdot M$, the trace is minimized when the M singular values are equal, $v_m = \sqrt{T_\tau}$. This gives $S = \sqrt{T_\tau}\Phi\Psi^\dagger$, and the factor $\Psi^\dagger$ can be absorbed into Φ without loss of generality. The resulting error covariance is

$$\mathrm{E}\left\{\tilde{h}_n \tilde{h}_n^\dagger\right\} = \left(\frac{M}{\rho T_\tau}\right) I_M. \tag{12}$$

To summarize, the duration of the training interval must be at least as great as the number of transmit antennas $T_\tau \geq M$, and the optimum training signals are of the form $S = \sqrt{T_\tau}\Phi$, where Φ is $T_\tau \times M$ unitary, i.e., $\Phi^\dagger \Phi = I_M$. The resulting ML estimate for the propagation matrix is unbiased, the performance is independent of the number of receive antennas, and the $M \cdot N$ errors are independent, and distributed as $\mathrm{CN}\left(0, \frac{M}{\rho T_\tau}\right)$.

5 Effects of Channel Estimation Error

Starting with its estimate for the propagation matrix, the receiver performs the QR factorization,

$$\hat{H} = \hat{R}\hat{Q} = \left(\hat{D} + \hat{U}\right)\hat{Q}. \tag{13}$$

5.1 Channel estimation errors, nulling, and cancellation

The nulling operation gives

$$Y_t = X_t \hat{Q}^\dagger . \tag{14}$$

Together, the nulling and cancellation operations are summarized by the left-hand side of (8), with Y_t according to (14), and with U and D replaced by $\hat{U}$ and $\hat{D}$ respectively. In turn this expression, by the successive substitution of X_t according to (1), the replacement of H by $\hat{H} - \tilde{H}$, and the use of the identity $\hat{H}\hat{Q}^\dagger = \hat{R} = \hat{D} + \hat{U}$, becomes

$$\begin{aligned}
&\left(\frac{\rho}{M}\right)^{-1/2} \left[X_t\hat{Q}^\dagger - (\rho/M)^{1/2}\hat{S}_t\hat{U}\right]\hat{D}^{-1} \\
&= \left(\frac{\rho}{M}\right)^{-1/2} \left[(\rho/M)^{1/2}S_tH\hat{Q}^\dagger + W_t\hat{Q}^\dagger - (\rho/M)^{1/2}\hat{S}_t\hat{U}\right]\hat{D}^{-1} \\
&= \left(\frac{\rho}{M}\right)^{-1/2} \left[(\rho/M)^{1/2}S_t\hat{H}\hat{Q}^\dagger - (\rho/M)^{1/2}S_t\tilde{H}\hat{Q}^\dagger + W_t\hat{Q}^\dagger - (\rho/M)^{1/2}\hat{S}_t\hat{U}\right]\hat{D}^{-1} \\
&= \left(\frac{\rho}{M}\right)^{-1/2} \left[(\rho/M)^{1/2}S_t\hat{D} + W_t\hat{Q}^\dagger - (\rho/M)^{1/2}S_t\tilde{H}\hat{Q}^\dagger + (\rho/M)^{1/2}\left(S_t - \hat{S}_t\right)\hat{U}\right]\hat{D}^{-1} \\
&= S_t + \left[(\rho/M)^{-1/2}W_t - S_t\tilde{H}\right]\hat{Q}^\dagger\hat{D}^{-1} + \left(S_t - \hat{S}_t\right)\hat{U}\hat{D}^{-1}.
\end{aligned} \tag{15}$$

This expression is exact, and it contains four terms. The first term is the desired signal S_t. As before we assume that $\hat{S}_t = S_t$, so the fourth term vanishes. The second term involves the receiver noise. The third term, containing $S_t\tilde{H}$, can be interpreted as crosstalk among the nominally decoupled virtual subchannels, due to the estimation error. The estimation errors that are associated with $\hat{D}$ and $\hat{Q}$ have only second-order effect in (15). Hence to first order, nulling and cancellation produces the following

$$\begin{aligned}
&\left(\frac{\rho}{M}\right)^{-1/2} \left[X_t\hat{Q}^\dagger - (\rho/M)^{1/2}\hat{S}_t\hat{U}\right]\hat{D}^{-1} \\
&\approx S_t + \left[(\rho/M)^{-1/2}W_t - S_t\tilde{H}\right]Q^\dagger D^{-1} \\
&= S_t + \left[(\rho/M)^{-1/2}\nu_t - S_t\tilde{H}Q^\dagger\right]D^{-1}.
\end{aligned} \tag{16}$$

5.2 Outage due to channel estimation error

An inspection of (16) discloses the possibility of two independent outage events, that arise from the independent terms D and $\tilde{H}$. The primary outage event is that the elements of D are too small to support the transmission rate. The secondary outage event is that the elements of the estimation error $\tilde{H}$ are big enough to produce a significant crosstalk term. The n-th component of the crosstalk is given by

$$\left[S_t\tilde{H}Q^\dagger\right]_n = \sum_{m=1}^{M} s_{tm}\left[\tilde{H}Q^\dagger\right]_{mn}, \tag{17}$$

and its variance, conditioned on $\tilde{H}$, is

$$\mathrm{E}\left\{\left|\left[S_t\tilde{H}Q^\dagger\right]_n\right|^2\right\} = \sum_{m=1}^{M}\left|\left[\tilde{H}Q^\dagger\right]_{mn}\right|^2, \tag{18}$$

where we used the facts that the s_{tm} are zero-mean, with unit variance, and that they are independent over m. The $M \times M$ matrix $\tilde{H}Q^\dagger$ has independent components that are zero-mean, complex Gaussian, with variance $M/(\rho T_\tau)$. Consequently the conditional variances of the M crosstalk terms are independent, identically distributed, and proportional to chi-square random variables with $2M$ degrees of freedom,

$$\left(\frac{2\rho T_\tau}{M}\right) \cdot \mathrm{E}\left\{\left|\left[S_t\tilde{H}Q^\dagger\right]_n\right|^2\right\} : \quad \chi^2_{2M}. \tag{19}$$

We say that a secondary outage event has occurred if any of the M conditional crosstalk variances is greater than some fraction β of the scaled receiver noise variance, where typically $\beta \leq 1$, or

$$\sup_n \mathrm{E}\left\{\left|\left[S_t\tilde{H}Q^\dagger\right]_n\right|^2\right\} > \frac{\beta M}{\rho}. \tag{20}$$

Utilizing (19) the outage probability is given in terms of the cumulative distribution for a chi-square random variable with $2M$ degrees of freedom by

$$P_{o2} = 1 - \left[P\left\{\chi^2_{2M} < 2\beta T_\tau\right\}\right]^M. \tag{21}$$

Note that the outage probability is independent of the expected SNR, and that it only depends on M and on the product βT_τ. Figure 2 displays the outage probability according to this formula, as a function of $\beta T_\tau/M$ for five values of M. The outage probability decays exponentially with the duration of the training interval, and the required training interval increases at a rate that is approximately linear with the number of transmit antennas.

6 Conclusions

The BLAST scheme does not impose especially stressful training requirements, and the operations of nulling and cancellation are robust with respect to errors in estimating the propagation matrix.

Our findings, notwithstanding, training emerges as a major issue when the total interval for training and for transmitting data is limited, either because of fast fading or because of a TDMA scenario, and where it is desired to maximize the effective transmission rate. We have seen that the required training interval grows approximately linearly with the number of transmit antennas. In turn Foschini [2] has shown that the capacity in units of bits per symbol is approximately proportional to the number of transmit antennas. Thus $T_\tau = \alpha M$, for some constant α, and this leave $T_d = T - T_\tau = T - \alpha M$ symbols for sending the message. The capacity is $C = \gamma M$ for some constant γ, so the total number of message bits that can be sent is equal to $CT_d = \gamma M(T - \alpha M)$. The optimum number of transmit antennas from the standpoint of maximizing the number of message bits is $M = \frac{T}{2\alpha}$, implying that, whatever the values of α and γ, the training interval is $T_\tau = T/2$. In other words, when trying to achieve the maximum possible throughput, half of the available interval is used for training.

Ideally one would like to achieve BLAST-like transmission rates with multiple antennas, while completely avoiding training and channel estimation. Some steps in this

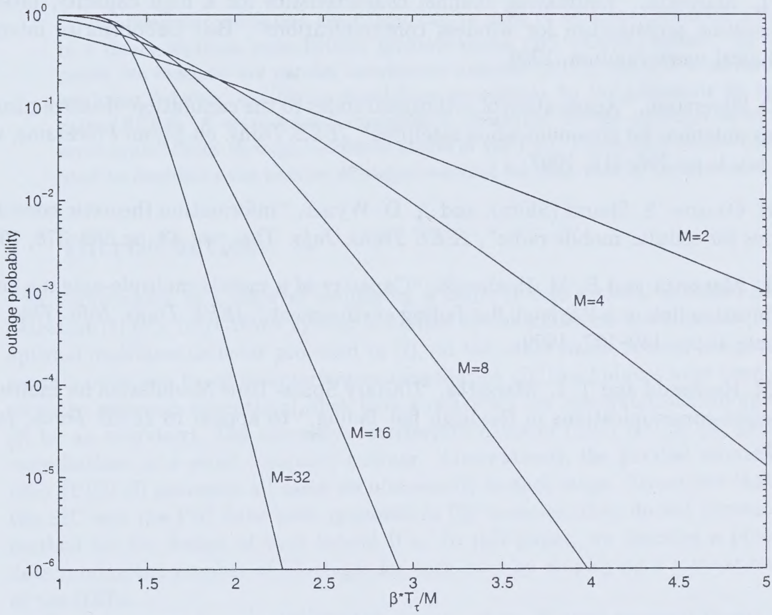


Figure 2: Estimation error-induced outage probability as a function of $\beta T_\tau / M$

direction are described in [6] and [7].

References

[1] I. E. Telatar, "Capacity of multi-antenna Gaussian channels," AT&T Bell Laboratories internal Technical Memorandum, 1995.

[2] G. J. Foschini, "Layered space-time architecture for wireless communication in a fading environment when using multi-element antennas," *Bell Labs. Tech. J.*, vol. 1, no. 2, pp. 41–59, 1996.

[3] T. L. Marzetta, "Estimating channel characteristics for a high capacity, layered space-time architecture for wireless communications", Bell Laboratories internal technical memorandum, 1996.

[4] S. D. Silverstein, "Application of orthogonal codes to the calibration of active phased array antennas for communication satellites", *IEEE Trans. on Signal Processing*, vol. 45, no. 1, pp.206–218, 1997.

[5] L. H. Ozarow, S. Shami (Shitz), and A. D. Wyner, "Information theoretic considerations for cellular mobile radio", *IEEE Trans. Info. Thy.*, vol 43, pp.359–378, 1994.

[6] T. L. Marzetta and B. M. Hochwald, "Capacity of a mobile multiple-antenna communication link in a Rayleigh flat-fading environment," *IEEE Trans. Info. Thy.*, vol 45, no. 1, pp. 139–157, 1999.

[7] B. M. Hochwald and T. L. Marzetta, "Unitary Space-Time Modulation for multiple-antenna communications in Rayleigh flat fading," to appear in *IEEE Trans. Info. Thy.*

Analysis of the Parallel Interference Canceler for DS/CDMA Signals *

Rajiv Chandrasekaran and John J. Shynk
Department of Electrical and Computer Engineering
University of California
Santa Barbara, CA 93106
shynk@ece.ucsb.edu

Abstract

Interference cancellation algorithms can be used to separate cochannel signals in a direct-sequence code-division multiple-access (DS/CDMA) system. In this paper, we examine the parallel interference canceler (PIC), which is a multistage nonlinear detector, and derive closed-form expressions for the achievable bit error rates (BERs). A state-space formulation is presented for the propagation of the error probabilities through successive stages of the PIC. These BER results can be used to determine the number of stages required for each user in the system.

1 Introduction

The performance of a receiver employing a bank of conventional matched-filter (MF) detectors [1] in a DS/CDMA system is limited by the multiuser interference (MUI). The optimal multiuser detector proposed in [2], on the other hand, is very complex. Consequently, receivers based on interference cancellation (IC) techniques have been proposed recently; although suboptimal, they have a relatively low computational complexity (see [3] for an overview). The successive interference canceler (SIC) [4], [5], [6], performs the cancellations in a serial (cascade) manner. Alternatively, the parallel interference canceler (PIC) [7] processes all users simultaneously in each stage. Structures that combine the SIC and the PIC have been proposed in [8]; however, they do not present a formal method for the design of such hybrid ICs. In this paper, we describe a procedure for determining the number of IC stages for each user by employing a state-space analysis of the BERs.

Consider a synchronous DS/CDMA system where N users transmit binary data over an additive white Gaussian noise (AWGN) channel. The received signal in the symbol interval $0 \leq t \leq T$ can be modeled as

$$r(t) = \sum_{i=1}^{N} A_i d_i s_i(t) + n(t) \tag{1}$$

where $\{A_i\}$ are the signal amplitudes, $\{s_i(t)\}$ are the signature waveforms, $\{d_i\}$ are the binary data with $P(d_i = 1) = P(d_i = -1) = 1/2$, and $n(t)$ is zero-mean AWGN with

---

*This work was supported by Applied Signal Technology, Inc., Fujant, Inc., and the University of California MICRO Program.

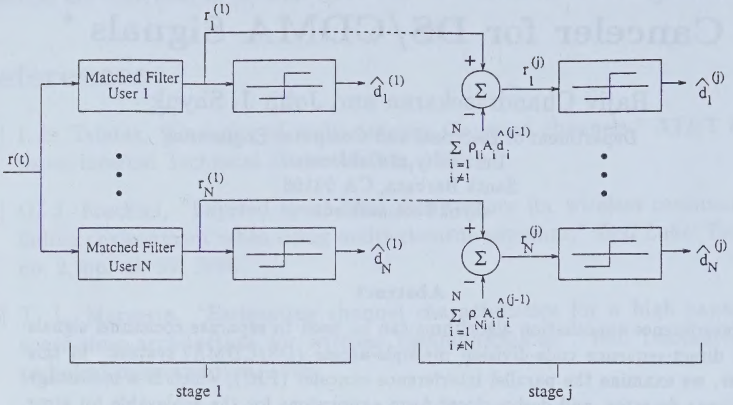


Figure 1: Block diagram of the PIC.

variance σ^2. The signature waveforms $\{s_i(t)\}$ are restricted to the time interval $[0, T)$ and have unit energy. It is assumed that the data bits generated by the different users are uncorrelated with each other and with the noise.

The PIC is a multistage nonlinear receiver with the architecture shown in Figure 1. For convenience in the analyses below, we assume that exact amplitude estimates for all users are available at the receiver. In the first stage, the data bits are estimated using filters that are matched to each of the users. If we define the cross-correlation coefficients

$$\rho_{ij} \triangleq \int_0^T s_i(t)s_j(t)dt, \qquad i,j = 1,\ldots,N, \tag{2}$$

then the decision variables at the output of the MF stage can be written as

$$r_i^{(1)} = A_i d_i + \sum_{k=1,k\neq i}^{N} \rho_{ik} A_k d_k + n_i, \qquad i = 1,\ldots,N. \tag{3}$$

Note that $\mathbf{n} = [n_1,\ldots,n_N]^T$ is a zero-mean Gaussian random vector whose correlation matrix contains the following elements: $\mathbf{R}_{ij} = \rho_{ij}\sigma^2$ for $i,j = 1,\ldots,N$. The corresponding estimates of the data bits are given by

$$\hat{d}_i^{(1)} = \mathrm{sgn}[r_i^{(1)}], \qquad i = 1,\ldots,N \tag{4}$$

where the superscript $^{(1)}$ denotes the first stage of the PIC.

In the succeeding stages of the PIC, estimates of the data bits are refined by subtracting the estimates from the previous stage. The motivation behind the successive refinement strategy is to improve the BER with each new stage. If the estimates from a given stage of the receiver are sufficiently reliable, then they can be used to cancel the MUI from the current decision variables, and thus reduce the BER in the next stage. Using this procedure, the decision variables in the mth stage can be written as follows:

$$r_i^{(m)} = A_i d_i + \sum_{k=1,k\neq i}^{N} \rho_{ik} A_k (d_k - \hat{d}_k^{(m-1)}) + n_i, \qquad i = 1,\ldots,N, \tag{5}$$

and the corresponding decoded bits are

$$\hat{d}_i^{(m)} = \mathrm{sgn}[r_i^{(m)}], \qquad i = 1, \ldots, N. \tag{6}$$

2 BER Analysis

2.1 Stage 1: Exact Analysis

Recall that the first stage of the PIC receiver is a bank of matched filters. Consider the estimate of the ith user bit in the first stage given by (3) and (4), i.e.,

$$\hat{d}_i^{(1)} = \mathrm{sgn}\left[A_i d_i + \sum_{k=1, k\neq i}^{N} \rho_{ik} A_k d_k + n_i \right]. \tag{7}$$

The probability of error in decoding the ith user in the first stage is

$$\begin{aligned} P_{e_i}^{(1)} &= P(d_i \neq \hat{d}_i^{(1)}) \\ &= \sum_{d_1=-1,1} \sum_{d_2=-1,1} \cdots \sum_{d_N=-1,1} P(d_i \neq \hat{d}_i^{(1)} | d_1, d_2, \ldots, d_N) P(d_1, d_2, \ldots, d_N) \\ &= \frac{1}{2^N} \sum_{d_1=-1,1} \sum_{d_2=-1,1} \cdots \sum_{d_N=-1,1} P(d_i \neq \hat{d}_i^{(1)} | d_1, d_2, \ldots, d_N) \end{aligned} \tag{8}$$

where we have used the fact that the user bits are independent and equiprobable. The expression in (8) can be evaluated by observing that

$$\begin{aligned} &P(d_i \neq \hat{d}_i^{(1)} | d_1, d_2, \ldots, d_N) \\ &\qquad = \begin{cases} P(n_i < -A_i - \sum_{k=1, k\neq i}^{N} \rho_{ik} A_k d_k | d_1, d_2, \ldots, d_N), & d_i = 1 \\ P(n_i > A_i - \sum_{k=1, k\neq i}^{N} \rho_{ik} A_k d_k | d_1, d_2, \ldots, d_N), & d_i = -1. \end{cases} \end{aligned} \tag{9}$$

Since the noise variables $\{n_i\}$ and the data bits $\{d_i\}$ are independent, we can rewrite (9) as follows:

$$\begin{aligned} P(d_i \neq \hat{d}_i^{(1)} | d_1, d_2, \ldots, d_N) &= \begin{cases} P(n_i < -A_i - \sum_{k=1, k\neq i}^{N} \rho_{ik} A_k d_k), & d_i = 1 \\ P(n_i > A_i - \sum_{k=1, k\neq i}^{N} \rho_{ik} A_k d_k), & d_i = -1 \end{cases} \\ &= \begin{cases} Q\left((A_i + \sum_{k=1, k\neq i}^{N} \rho_{ik} A_k d_k)/\sigma \right), & d_i = 1 \\ Q\left((A_i - \sum_{k=1, k\neq i}^{N} \rho_{ik} A_k d_k)/\sigma \right), & d_i = -1 \end{cases} \end{aligned} \tag{10}$$

where $Q(x) \triangleq (1/\sqrt{2\pi}) \int_x^\infty e^{-t^2/2} dt$ [1]. Finally, inserting (10) into (8) yields the probability of error for the ith user in the MF stage.

2.2 Stage 2: Exact Analysis

The estimates of the data bits from the MF stage are used to cancel the MUI in the second stage. Assume that we are interested in estimating $\{d_i\}$, the bits of the ith user. The corresponding decision variable is (from (5) with $m = 2$)

$$r_i^{(2)} = A_i d_i + \sum_{k=1,k\neq i}^{N} \rho_{ik} A_k e_k^{(1)} + n_i, \tag{11}$$

where $e_k^{(1)} \triangleq d_k - \hat{d}_k^{(1)}$. The resulting decision is given by $\hat{d}_i^{(2)} = \operatorname{sgn}[r_i^{(2)}]$ (from (6) with $m = 2$).

Define the error vector $\mathbf{e}_i^{(1)} \triangleq [e_1^{(1)}, \ldots, e_{i-1}^{(1)}, e_{i+1}^{(1)}, \ldots, e_N^{(1)}]^T$ and the data vector $\mathbf{d} \triangleq [d_1, \ldots, d_N]^T$. Note that since the data are binary, each $e_k^{(1)} \in \{-2, 0, 2\}$. Hence, the random vector $\mathbf{e}_i^{(1)}$ can take on one of $3^{(N-1)}$ possible values, and the data vector $\mathbf{d}$ can take on one of 2^N possible values. The probability of error for the ith user in the second stage can be written as follows:

$$P_{e_i}^{(2)} = P(d_i \neq \hat{d}_i^{(2)}) = \sum_{\mathbf{d}} \sum_{\mathbf{e}_i^{(1)}} P(d_i \neq \hat{d}_i^{(2)} | \mathbf{d}, \mathbf{e}_i^{(1)}) P(\mathbf{d}, \mathbf{e}_i^{(1)}) \tag{12}$$

where the summations are over all possible values of $\mathbf{e}_i^{(1)}$ and $\mathbf{d}$. Although there are $2^N 3^{(N-1)}$ terms in (12), some are zero because $e_k^{(1)}$ can never be equal to 2 (or -2) if $d_k = -1$ (or 1). As a result, there are only $2^N 2^{(N-1)}$ nonzero terms. From the right-hand side of (12), we can write

$$P(d_i \neq \hat{d}_i^{(2)} | \mathbf{d}, \mathbf{e}_i^{(1)}) = \begin{cases} P(r_i^{(2)} < 0 | \mathbf{d}, \mathbf{e}_i^{(1)}), & d_i = 1 \\ P(r_i^{(2)} > 0 | \mathbf{d}, \mathbf{e}_i^{(1)}), & d_i = -1. \end{cases} \tag{13}$$

Substituting (11) for $r_i^{(2)}$ yields

$$\begin{aligned} P(d_i \neq \hat{d}_i^{(2)} | \mathbf{d}, \mathbf{e}_i^{(1)}) &= \begin{cases} P(n_i < -A_i - \sum_{k=1,k\neq i}^{N} \rho_{ik} A_k e_k^{(1)} | \mathbf{d}, \mathbf{e}_i^{(1)}), & d_i = 1 \\ P(n_i > A_i - \sum_{k=1,k\neq i}^{N} \rho_{ik} A_k e_k^{(1)} | \mathbf{d}, \mathbf{e}_i^{(1)}), & d_i = -1 \end{cases} \\ &= P(n_i > A_i + d_i \sum_{k=1,k\neq i}^{N} \rho_{ik} A_k e_k^{(1)} | \mathbf{d}, \mathbf{e}_i^{(1)}) \end{aligned} \tag{14}$$

where we have utilized the fact that the Gaussian pdf is symmetric about zero. The analysis proceeds in a manner similar to that used for the first stage, although it is more complicated. Only the results are presented in this paper; the reader is referred to [9] for additional details.

Consider the regions $\{R_j^{(1)} | j = 1, \ldots, N \text{ and } j \neq i\}$ defined by

$$R_j^{(1)} \triangleq \begin{cases} [A_j + d_j \sum_{k=1,k\neq j}^{N} \rho_{jk} A_k e_k^{(1)}, \infty), & \{e_j^{(1)} = 2 \text{ and } d_j = 1\} \text{ or} \\ & \{e_j^{(1)} = -2 \text{ and } d_j = -1\} \\ (-\infty, A_j + d_j \sum_{k=1,k\neq j}^{N} \rho_{jk} A_k e_k^{(1)}), & e_j^{(1)} = 0 \end{cases} \tag{15}$$

and $R_i^{(1)} \triangleq [A_i + d_i \sum_{k=1,k\neq i}^{N} \rho_{ik} A_k e_k^{(1)}, \infty)$ for $j = i$. Then (14) can be rewritten as an integral of the N-dimensional joint Gaussian pdf $f(n_1, n_2, \ldots, n_N)$ over a volume, i.e.,

$$P(d_i \neq \hat{d}_i^{(2)} | \mathrm{d}, \mathrm{e}_i^{(1)}) = \frac{\int_{\mathbf{V}} f(n_1, \ldots, n_N) dn_1 \cdots dn_N}{P(\mathrm{e}_i^{(1)} | \mathrm{d})} \tag{16}$$

where the volume $\mathbf{V}$ corresponds to the intersection of the regions defined above. We thus have that

$$\begin{aligned} P(d_i \neq \hat{d}_i^{(2)} | \mathrm{d}, \mathrm{e}_i^{(1)}) P(\mathrm{d}, \mathrm{e}_i^{(1)}) &= \frac{\int_{\mathbf{V}} f(n_1, \ldots, n_N) dn_1 \cdots dn_N}{P(\mathrm{e}_i^{(1)} | \mathrm{d})} P(\mathrm{d}, \mathrm{e}_i^{(1)}) \\ &= \int_{\mathbf{V}} f(n_1, \ldots, n_N) dn_1 \cdots dn_N P(\mathrm{d}) \\ &= \frac{1}{2^N} \int_{\mathbf{V}} f(n_1, \ldots, n_N) dn_1 \ldots dn_N. \end{aligned} \tag{17}$$

Combining (17) and (12) yields the desired result for $P_{e_i}^{(2)}$.

2.3 Stage $j-1$ to Stage j: Approximate Analysis

The BER analysis for the second stage yields integrals over volumes of the N-dimensional joint Gaussian pdf. It is clear that for the succeeding stages, the exact analysis will be even more complicated. Thus, in this section we develop an approximate analysis that estimates the BER in the jth stage of the receiver as a function of the BER in stage $j-1$.

The probability of error in detecting the ith user in the jth stage of the PIC can be written as follows:

$$P_{e_i}^{(j)} = P(\hat{d}_i^{(j)} \neq d_i) = \sum_{m=1}^{2^{N-1}} P(\hat{d}_i^{(j)} \neq d_i | E_m) P(E_m) \tag{18}$$

where the events $\{E_m,\ m = 1, \ldots, 2^{N-1}\}$ are defined as

$$E_m \triangleq \{\hat{d}_1^{(j-1)} *_1 d_1, \ldots, \hat{d}_{i-1}^{(j-1)} *_{i-1} d_{i-1}, \hat{d}_{i+1}^{(j-1)} *_i d_{i+1}, \ldots, \hat{d}_N^{(j-1)} *_{N-1} d_N\}. \tag{19}$$

Each of the operators $\{*_1, \ldots, *_{N-1}\}$ can be either "$=$" or "$\neq$" (not to be confused with the convolution operator), resulting in 2^{N-1} combinations. We assume that the decisions made on the N users in stage $j-1$ are independent, i.e., the events $\{\hat{d}_1^{(j-1)} *_1 d_1\}, \ldots, \{\hat{d}_{i-1}^{(j-1)} *_{i-1} d_{i-1}\}, \{\hat{d}_{i+1}^{(j-1)} *_i d_{i+1}\}, \ldots, \{\hat{d}_N^{(j-1)} *_{N-1} d_N\}$ are independent so that

$$P(E_m) = P(\hat{d}_1^{(j-1)} *_1 d_1) \cdots P(\hat{d}_{i-1}^{(j-1)} *_{i-1} d_{i-1}) P(\hat{d}_{i+1}^{(j-1)} *_i d_{i+1}) \cdots P(\hat{d}_N^{(j-1)} *_{N-1} d_N) \tag{20}$$

where for $m = 1, \ldots, i-1$:

$$P(\hat{d}_m^{(j-1)} *_m d_m) = \begin{cases} P_{e_m}^{(j-1)}, & \text{if } *_m \text{ is } \neq \\ 1 - P_{e_m}^{(j-1)}, & \text{if } *_m \text{ is } =, \end{cases} \tag{21}$$

and for $m = i+1, \ldots, N$:

$$P(\hat{d}_m^{(j-1)} *_{m-1} d_m) = \begin{cases} P_{e_m}^{(j-1)}, & \text{if } *_{m-1} \text{ is } \neq \\ 1 - P_{e_m}^{(j-1)}, & \text{if } *_{m-1} \text{ is } =. \end{cases} \tag{22}$$

Equation (18) can be simplified by assuming that the error probabilities in stage $j-1$ are small, so that only the terms which are linear in $\{P_{e_i}^{(j-1)}\}$ are retained. Thus,

$$P_{ei}^{(j)} \approx (1 - \sum_{m=1}^{N} P_{em}^{(j-1)})\gamma_i + \sum_{m=1, m\neq i}^{N} P_{em}^{(j-1)}\alpha_{im}, \quad i = 1, \ldots, N \tag{23}$$

where $\{\gamma_i,\ i = 1, \ldots, N\}$, and $\{\alpha_{im},\ i, m = 1, \ldots, N \text{ and } i \neq m\}$, are evaluated as follows:

$$\begin{aligned} \gamma_i &= P(\hat{d}_i^{(j-1)} \neq d_i | \hat{d}_1^{(j-1)} = d_1, \ldots, \hat{d}_{i-1}^{(j-1)} = d_{i-1}, \hat{d}_{i+1}^{(j-1)} = d_{i+1}, \ldots, \hat{d}_N^{(j-1)} = d_N) \\ &= P(\text{sgn}[A_i d_i + n_i] \neq d_i) = Q(A_i/\sigma) \end{aligned} \tag{24}$$

and

$$\begin{aligned} \alpha_{im} &= P(\hat{d}_i^{(j-1)} \neq d_i | \hat{d}_1^{(j-1)} = d_1, \ldots, \hat{d}_m^{(j-1)} \neq d_m, \\ &\qquad \ldots, \hat{d}_{i-1}^{(j-1)} = d_{i-1}, \hat{d}_{i+1}^{(j-1)} = d_{i+1}, \ldots, \hat{d}_N^{(j-1)} = d_N) \\ &= P(\text{sgn}[A_i d_i + \rho_{im} A_m (d_m - \hat{d}_m^{(j-1)}) + n_i] \neq d_i) \\ &= \frac{1}{2} Q((A_i - 2\rho_{im} A_m)/\sigma) + \frac{1}{2} Q((A_i + 2\rho_{im} A_m)/\sigma) \end{aligned} \tag{25}$$

where we have dropped the conditioning based on the assumption that the decisions made on the N users are independent.

From (23), (24), and (25), the following recursion is obtained:

$$\mathbf{P}_e^{(j)} = \mathbf{A}\mathbf{P}_e^{(j-1)} + \mathbf{u} \tag{26}$$

where $\mathbf{P}_e^{(j)} \triangleq [P_{e1}^{(j)}, \ldots, P_{eN}^{(j)}]^T$, $\mathbf{u} \triangleq [Q(A_1/\sigma), \ldots, Q(A_N/\sigma)]^T$, and the (i, m)th element of the $N \times N$ matrix $\mathbf{A}$ is given by

$$\mathbf{A}_{i,m} = \begin{cases} 0, & i = m \\ \frac{1}{2} Q((A_i - 2\rho_{im} A_m)/\sigma) + \frac{1}{2} Q((A_i + 2\rho_{im} A_m)/\sigma) - Q(A_m/\sigma), & \text{otherwise.} \end{cases} \tag{27}$$

Note that due to the approximation in (23), the error probabilities given by (26) could be negative. However, this does not affect the accuracy of the results because such cases arise only when the error probability for a specific user in a given stage is very close to zero. This implies that the error probabilities for such users have converged; consequently, adding more PIC stages will not significantly improve the error probabilities. Since the analysis is approximate, we cannot expect to estimate very low probabilities of errors in such cases. Thus, we constrain the error probabilities to be nonnegative by forcing any negative probabilities to be zero.

3 State-Space Analysis

In this section, a state-space model for the propagation of the error probabilities through successive stages of the PIC is investigated. We describe how these results can be used to determine the number of IC stages required for each user in the system.

Recall that (26) describes how the error probabilities propagate from stage $j-1$ to stage j. Consider an eigen-decomposition of the matrix $\mathbf{A}$ as follows: $\mathbf{AV} = \mathbf{V\Lambda}$ where $\mathbf{V} \triangleq [\mathbf{v}_1, \ldots, \mathbf{v}_N]$ is a matrix whose columns contain the eigenvectors of $\mathbf{A}$, and $\mathbf{\Lambda}$ is a

1	−0.06	−0.13	−0.06	0.25	0.06	−0.19	0.25	0	0
−0.06	1	0.19	0	0.06	0	−0.13	0.31	−0.19	−0.06
−0.13	0.19	1	0.31	0.13	0.19	0.06	−0.13	0	0
−0.06	0	0.31	1	0.19	−0.13	−0.25	−0.31	0.19	0.19
0.25	0.06	0.13	0.19	1	−0.06	0.06	0	0.25	0.13
0.06	0	0.19	−0.13	−0.06	1	−0.13	−0.06	0.19	−0.06
−0.19	−0.13	0.06	−0.25	0.06	−0.13	1	−0.06	−0.44	0.06
0.25	0.31	−0.13	−0.31	0	−0.06	−0.06	1	−0.38	0
0	−0.19	0	0.19	0.25	0.19	−0.44	−0.38	1	0.13
0	−0.06	0	0.19	0.13	−0.06	0.06	0	0.13	1

Table 1: Correlation Values ρ_{ij} of the Spreading Codes for a Ten-User DS/CDMA System

diagonal matrix whose ith diagonal element is the eigenvalue λ_i corresponding to the eigenvector $\mathbf{v}_i$. The transformation $\mathbf{r}^{(j)} \triangleq \mathbf{V}^{-1}\mathbf{P}_e{}^{(j)}$ on (26) yields

$$\mathbf{r}^{(j)} = \mathbf{V}^{-1}\mathbf{A}\mathbf{P}_e^{(j-1)} + \mathbf{V}^{-1}\mathbf{u} = \mathbf{\Lambda}\mathbf{r}^{(j-1)} + \mathbf{\Gamma} \tag{28}$$

where the eigen-decomposition of $\mathbf{A}$ has been employed and $\mathbf{\Gamma} \triangleq \mathbf{V}^{-1}\mathbf{u}$. Equation (28) can be decomposed into a set of N independent modes; each mode is described by a scalar difference equation as follows:

$$r_i^{(j)} = \lambda_i r_i^{(j-1)} + \gamma_i, \qquad i = 1, \ldots, N \tag{29}$$

where $\mathbf{r} \triangleq [r_1, \ldots, r_N]^T$ and $\mathbf{\Gamma} \triangleq [\gamma_1, \ldots, \gamma_N]^T$. This recursion can be explicitly solved, yielding

$$r_i^{(j)} = \lambda_i^{j-1} r_i^{(1)} + \gamma_i \sum_{k=0}^{j-2} \lambda_i^k, \qquad i = 1, \ldots, N. \tag{30}$$

Clearly, if all the eigenvalues $\{\lambda_i\}$ are within the unit circle, then the system is stable and the modes converge such that

$$r_i^{(\infty)} = \frac{\gamma_i}{1 - \lambda_i}, \qquad i = 1, \ldots, N. \tag{31}$$

Furthermore, the number of iterations m_i approximately required for convergence of the ith mode can be estimated by finding m_i such that $r_i^{(m_i)} \approx \gamma_i$. Observe in (30) that if $|\lambda_i| \ll 1$ (which is usually the case for $\mathbf{A}$), then

$$\gamma_i \sum_{k=0}^{j-2} \lambda_i^k \approx \gamma_i, \qquad i = 1, \ldots, N. \tag{32}$$

Thus, the number of iterations required for convergence of the ith mode can be estimated by finding the m_i that satisfies $|\lambda_i^{m_i - 1} r_i^{(1)}| \approx |\gamma_i|$, i.e., the m_i at which the first term in the right-hand side of (30) is approximately equal to the second term. For the subsequent stages, the contribution of the first term starts to become negligible relative to the second term. This results in

$$m_i = 1 + \mathrm{round}\left(\frac{\log|\gamma_i / r_i^{(1)}|}{\log|\lambda_i|}\right), \qquad i = 1, \ldots, N \tag{33}$$

where the function round(α) rounds α to the nearest integer.

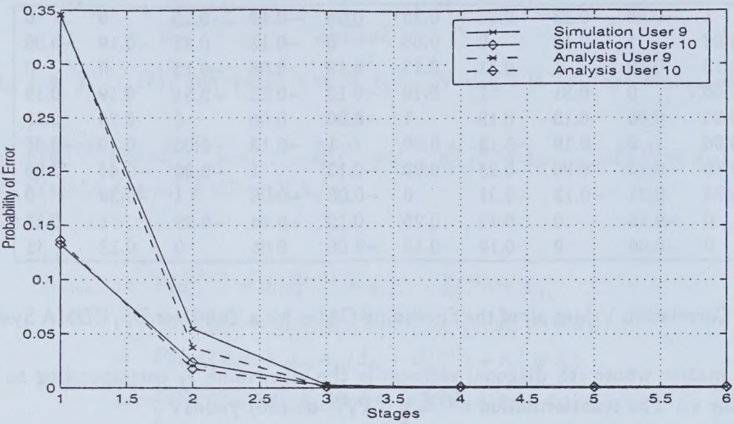


Figure 2: Analysis and simulation BER results for the two weak users.

4 Computer Simulations

We consider a DS/CDMA system with ten users and spreading codes of length 64. Ten million events were generated using Monte-Carlo simulations to estimate the error probabilities. As a result, estimates of the error probabilities on the order of 10^{-5} (or higher) are quite reliable [10]. Table 1 shows the correlations between the spreading codes; note that most are in the range $(-0.2, 0.2)$, so that the noise variables are approximately uncorrelated.

The power levels of the ten users were chosen to simulate a near-far scenario: the SNR for each of the eight strong users was 20 dB and that for each of the two weak users was 10 dB. Figure 2 shows the BER performance for the two weak users. Observe that the analytical results closely match those of the computer simulations. In Figure 3, the accuracy of the analysis is also verified for four of the strong users (the other strong users have similar results). As expected, the weak users require more PIC stages (three stages) than the strong users (two stages). Note that even though the strong users have the same SNR, their BERs in the first stage differ because of the different correlations between the noise variables. In the succeeding stages of the PIC, the BERs are determined primarily by the amplitude levels, i.e., the BERs of the strong users are of the same order, and those of the weak users are also of similar order.

5 IC Receiver Design

For the preceding ten-user system, the state-space analysis demonstrates that the system is stable because $|\lambda_i| < 1$; the absolute values of the eigenvalues are given by 4.93×10^{-2}, 4.93×10^{-2}, 3.66×10^{-3}, 7.15×10^{-5}, 3.28×10^{-5}, 2.89×10^{-5}, 2.73×10^{-5}, 1.43×10^{-7}, 8.48×10^{-8}, and 3.90×10^{-13}. Decomposing the system into the ten independent modes, (33) indicates that four modes require three stages to converge, whereas the rest of the modes converge in only two stages. This provides us with an estimate of the number of stages required for each of the users. These results are quite close to that observed in

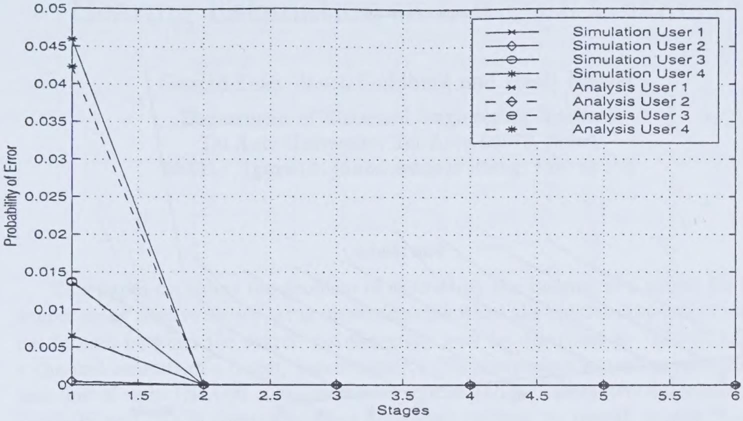


Figure 3: Analysis and simulation BER results for strong users $1 - 4$.

the preceding computer simulations.

In order to design the IC-based receiver, consider the "waterfall" plot of the error probabilities shown in Figure 4. From these results, we conclude that users 9 and 10 require three stages, users 1, 3, 4, and 8 require two stages, and the remaining users require just one stage. Hence, (33) yields a conservative estimate of the number of stages required.

6 Conclusion

In this paper, we have developed an IC-based receiver design strategy for DS/CDMA signals. Closed-form analytical expressions were derived for the parallel interference canceler (PIC), yielding exact results for the BERs achieved in the first two stages. We also described an approximate analysis for the BER obtained in any stage. Computer simulations were presented to demonstrate the accuracy of the analytical models, from which it can be concluded that the PIC performance depends more heavily on the relative amplitude levels than on the cross-correlations of the signature waveforms. A state-space analysis illustrated the BER convergence properties in successive stages of the PIC. These analytical results can be used to determine the number of stages required in the interference canceler for each user.

References

[1] J. G. Proakis, *Digital Communications.* New York: McGraw-Hill, third ed., 1995.

[2] S. Verdu, "Minimum probability of error for asynchronous Gaussian multiple-access channels," *IEEE Trans. on Information Theory*, vol. IT-32, pp. 85–96, Jan. 1986.

[3] S. Moshavi, "Multi-user detection for DS-CDMA communications," *IEEE Communications Mag.*, vol. 34, pp. 124–136, Oct. 1996.

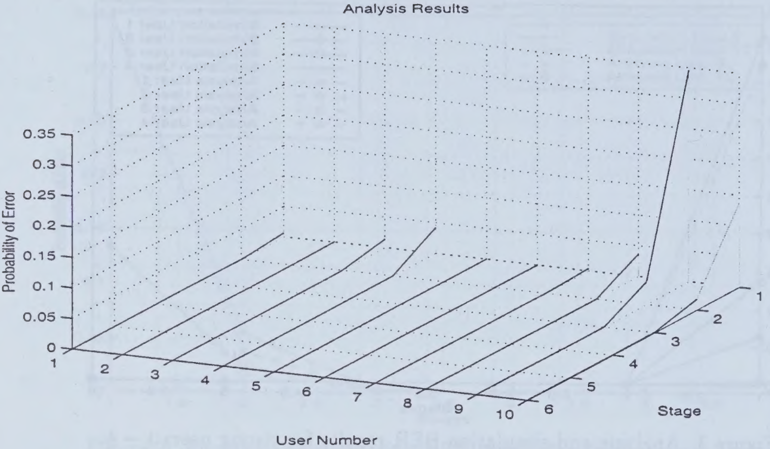


Figure 4: Error probabilities for each user based on the state-space analysis.

[4] A. J. Viterbi, "Very low rate convolutional codes for maximum theoretical performance of spread-spectrum multiple-access channels," *IEEE Journal on Selected Areas in Communications*, vol. 8, pp. 641–649, May 1990.

[5] P. Dent, B. Gudmundson, and M. Ewerbring, "CDMA-IC: A novel code division multiple access scheme based on interference cancellation," in *Proc. Third IEEE Int. Symp. on Personal, Indoor, and Mobile Radio Communications*, (Boston, MA), pp. 98–102, Oct. 1992.

[6] P. R. Patel and J. M. Holtzman, "Analysis of a DS/CDMA successive interference cancellation scheme using correlations," in *Proc. IEEE Global Telecommunications Conf.*, (Houston, TX), pp. 76–80, Nov. 1993.

[7] M. K. Varanasi and B. Aazhang, "Multistage detection in asynchronous code-division multiple-access communications," *IEEE Trans. on Communications*, vol. 38, pp. 509–519, Apr. 1990.

[8] D. Koulakiotis and A. H. Aghvami, "Evaluation of a DS/CDMA multiuser receiver employing a hybrid form of interference cancellation in Rayleigh-fading channels," *IEEE Communications Letters*, vol. 2, pp. 61–63, Mar. 1998.

[9] K.-C. Lai, R. Chandrasekaran, R. E. Cagley, and J. J. Shynk, "Multistage interference cancellation algorithms for DS/CDMA signals." In preparation.

[10] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes in C.* Cambridge University Press, 1995.

Bearing Estimation in a Ricean Channel

Garold Fuks, Jason Goldberg and Hagit Messer

Department of Electrical Engineering–Systems
Tel Aviv University, Tel Aviv 69978, Israel
EMAIL: {garold,jason,messer}@eng.tau.ac.il

Abstract

This paper considers the problem of estimating the bearing of a single, far field source using passive sensor array measurements when the spatial propagation channel formed between the source and the array may be described as "Ricean." Such a channel consists of a direct, line of sight (LOS) component as well as an indirect, non line of sight (NLOS) component due to scattering. Parametric descriptions of the LOS and NLOS components are presented yielding an overall spatial channel description which is more realistic than previously proposed models. The associated source bearing estimation problem is formulated, and the Cramer Rao Lower Bound (CRLB) is evaluated in order to study relationship between the Ricean problem to the better known LOS only and NLOS only problems. The performance of previously proposed nonlinear least squares (LS) and weighted least squares (WLS) procedures as well as the computationally simpler minimum variance and conventional beamformers (BF's) are compared to the CRLB for a wide variety of scenarios. It is found that the beamforming procedures often offer performance comparable to that of the far more complicated LS and WLS procedures.

1 Introduction

Classical source bearing estimation research has typically focused on sources which are modeled as a single point in space giving rise to LOS propagation conditions, e.g., [1]-[2]. More recently, so-called "scattered" source models have been proposed for applications where local scatters about the source interact with the transmitted signal before impinging on the array, thus resulting in NLOS propagation. Such applications include mobile communications e.g., [3], sonar e.g., [4], and radar e.g., [5]. Various techniques for the estimation of the bearing of a scattered source have been considered in the literature based on parametric spatial channel models, e.g., [6]-[10].

This paper addresses the bearing estimation problem when the propagation conditions are such that both a LOS and a NLOS component are present. Here, since (as will be seen) the LOS channel component is modeled as deterministic and the NLOS component is modeled as a complex Gaussian random process, the overall channel may be described as "Ricean" because its magnitude is described by a Ricean probability density function. A parametric Ricean spatial channel model has been previously proposed in the literature [11]. The model proposed in this paper is more accurate since it explicitly includes a Doppler shift on the LOS component due to source motion e.g., [12]. Also [11] presents bearing estimation procedures based on LS and WLS fits of the array sample correlation matrix both of which require computationally expensive two dimensional searches. As an alternative to such complex procedures, the performance of the conventional point source BF and the minimum variance BF (each of which requires a one dimensional search) is considered for Ricean sources. A simulation study is carried out to compare the performance of each of the techniques under a wide variety of scenarios. Conditions for which the performance of the beamforming techniques (i) give performance comparable to the LS and WLS methods and (ii) nearly achieve the CRLB are identified. In addition, a comparison of the LOS, NLOS, and Ricean source CRLB's is used to give insight into both the inherent limitations of the problem as well as the expected performance of some of the estimators.

2 The Ricean Channel Model

In the classical point source model, a single (LOS) wave-front path impinges on the array due to the transmitting source e.g., [2]. An extension of this simple model is that of a moving source possessing some Doppler shift (corresponding to a single ray version of the model in [13]).

Simple LOS models may not be sufficiently accurate in scenarios containing multipath due to local scattering about the source as is common, for example, in wireless communications in an urban environment e.g., [3]. For such scenarios, the Central Limit Theorem is typically invoked to model the sum of the contributions due to each scatterer at each sensor of the array as complex and Gaussian giving rise to so-called "Rayleigh fading" [14]. For sources of sufficiently narrow bandwidth relative to the channel's coherence bandwidth, the channel effectively manifests itself as multiplicative noise (the so-called "flat fading" assumption) e.g., [3], [14]. Moreover, for sampling instances which are spaced sufficiently wide relative to the channel's coherence time, the channel is temporally uncorrelated (corresponding to the so-called "incoherently distributed" (ID) source model of [6]). A parametric spatial density function is used to characterize the spatial extent of the source about its mean angle (which usually corresponds to source bearing) e.g., [6]-[8].

The Ricean channel considered in this paper is now mathematically formulated simply as a combination of the LOS and NLOS components just described. Consider a single narrow-band far field source centered at frequency, ω_c, emitting a signal of complex envelope, $s(t)$, which propagates via a Ricean channel and impinges on an array of M passive sensors. Under the flat fading assumption discussed above, the complex envelope of the $M \times 1$ noiseless array observation vector is written as:

$$\mathbf{x}(t_k) \;=\; \mathbf{h}(t_k)s(t_k) \qquad \mathbf{h}(t_k) = \alpha e^{j\phi} e^{j\omega_c t_k} \mathbf{a}(\theta_o) + \mathbf{b}(t_k)$$

where $\mathbf{a}(\theta_o)$ is an ideal point source steering vector for an array of omni-directional, unit gain sensors with $[\mathbf{a}(\theta_o)]_m = e^{-j\omega_c \tau_m(\theta_o)}$. The differential delays across the array are written as $\tau_m(\theta_o) = -\frac{1}{c}[x_m \cos(\theta_o) + y_m \sin(\theta_o)]$ where (x_m, y_m) denotes the location of the mth sensor in the plane, c is the propagation speed, and $[\cdot]_m$ denotes the mth element of a vector. ϕ represents the phase difference between the receiver and transmitter, ω_d corresponds to a Doppler shift of the carrier frequency due to the mobile movement. The (random) NLOS component vector is given by:

$$\mathbf{b}(t_k) \;=\; \int_{-\pi}^{\pi} f(\theta, t_k)\mathbf{a}(\theta)d\theta \tag{1}$$

where $f(\theta, t)$ is a complex random spatio-temporal weighting function. As mentioned earlier, the Central Limit Theorem enables $\mathbf{b}(t_k)$ to be modeled as zero mean, circular Gaussian random vector process. Under the ID source assumption, the NLOS component vectors are uncorrelated from snapshot to snapshot. Thus:

$$\begin{aligned} \mathbf{b}(t_k) &\sim CN(\mathbf{0}, \mathbf{R}_b(\Theta)) \\ \mathbf{R}_b(t_k - t_l/\Theta) &= E[\mathbf{b}(t_k)\mathbf{b}^H(t_l)] = \delta_{kl}\sigma_b^2 \int_{-\pi}^{\pi} p(\theta/\Theta)\mathbf{a}(\theta)\mathbf{a}^H(\theta)d\theta \end{aligned} \tag{2}$$

where $p(\theta/\Theta)$ is a nonnegative, unit area spatial weighting function over $\theta \in [-\pi, \pi)$, $E[\cdot]$ denotes expectation, and δ_{kl} denotes the Kronecker delta function. This weighting function is typically characterized by a simple parametric model such as the Gaussian or Uniform distribution parameterized by a mean angle and spatial spreading parameter $\Theta = [\theta_o, \Delta]^T$ e.g., [7]-[8].

The overall channel vector $\mathbf{h}(t_k)$ is the sum of the LOS and ID NLOS components and is therefore modeled as a complex Gaussian circular random process of specified time varying mean and time invariant covariance:

$$\begin{aligned} \mathbf{h}(t_k) &\sim CN(\mathbf{m}_h(t_k), \mathbf{C}_h(t_k)) \qquad \mathbf{m}_h(t_k) = E\{\mathbf{h}(t_k)\} = \alpha e^{j[\omega_d t_k + \phi]}\mathbf{a}(\theta_o) \\ \mathbf{C}_h(t_k) &= E\{[\mathbf{h}(t_k) - \mathbf{m}_h(t_k)][\mathbf{h}(t_k) - \mathbf{m}_h(t_k)]^H\} = E\{\mathbf{b}(t_k)\mathbf{b}^H(t_k)\} \\ \mathbf{C}_h(t_k, t_l) &= E[\mathbf{h}(t_k) - \mathbf{m}_h(t_k)][\mathbf{h}(t_l) - \mathbf{m}_h(t_l)]^H = \mathbf{C}_h(t_k)\delta_{kl} = \mathbf{C}_h\delta_{kl} = \sigma_b^2\mathbf{R}_b(\theta_o, \Delta)\delta_{kl}. \end{aligned} \tag{3}$$

3 Problem Formulation

Consider K noisy snapshot measurements of the wavefront generated by a source propagating through a Ricean channel and impinging on an M element array :

$$\mathbf{y}(t_k) = \mathbf{h}(t_k)s(t_k) + \mathbf{n}(t_k) \qquad k = 1 \dots K \tag{4}$$

The additive noise, $\mathbf{n}(t_k)$, is a zero mean, spatially and temporally white, stationary, complex circular Gaussian random vector process with covariance matrix $\sigma_n^2\mathbf{I}_M$, where $\mathbf{I}$ denotes identity matrix of a specified dimension. While not essential, the simplifying assumption of known, unit power transmitted signal will be made. (Such an assumption may be reasonable, for example, in wireless communications systems which use periodically transmitted training sequences.) Under these conditions the overall observation vector $\mathbf{z} = [\mathbf{y}^T(t_1) \dots \mathbf{y}^T(t_K)]^T$ is Gaussian with uncorrelated blocks, giving rise to an overall block diagonal correlation matrix with blocks:

$$\begin{aligned} \mathbf{y}(t_k) &\sim CN(\mathbf{m}_y(t_k), \mathbf{C}_y(t_k)) \\ \mathbf{m}_y(t_k) &= E\{\mathbf{y}(t_k)\} = \mathbf{m}_h(t_k)s(t_k) \\ \mathbf{C}_y(t_k, t_l) &= E\{[\mathbf{y}(t_k) - \mathbf{m}_y(t_k)][\mathbf{y}(t_l) - \mathbf{m}_y(t_l)]^H\} = (|s(t_k)|^2\mathbf{C}_h + \sigma_n^2\mathbf{I}_M)\delta_{kl} \end{aligned} \tag{5}$$

The "bearing estimation problem" for a source in a Ricean channel is defined in this paper as follows: given K snapshots $\{\mathbf{y}(t_k)\}, k = 1 \dots K$ and knowledge of the transmitted signal $s(t_k)$, estimate all the unknown deterministic parameters : $\chi = [\theta_o, \Delta, \sigma_b^2, \sigma_n^2, \alpha, \phi, \omega_d]^T$. The parameter of interest is the mean angle, θ_o, and the other parameters are nuisance parameters.

4 The CRLB

The CRLB on the estimation error of the unknown parameter vector $\chi = [\theta_o, \Delta, \sigma_b^2, \sigma_n^2, \alpha, \phi, \omega_d]^T$ is the inverse of the Fisher information matrix (FIM) with mn-th element: $J_{mn} = \sum_{k=1}^{K} \tilde{J}_{mn}(t_k)$ where the FIM elements for the snapshot at t_k are given by e.g., [15]:

$$\tilde{J}_{mn}(t_k) = tr\left\{\mathbf{C}_y^{-1}(t_k)\frac{\partial \mathbf{C}_y(t_k)}{\partial \chi_m}\mathbf{C}_y^{-1}(t_k)\frac{\partial \mathbf{C}_y(t_k)}{\partial \chi_n}\right\} + 2\Re\left\{\frac{\partial \mathbf{m}_y^H(t_k)}{\partial \chi_m}\mathbf{C}_y^{-1}(t_k)\frac{\partial \mathbf{m}_y(t_k)}{\partial \chi_n}\right\} \quad (6)$$

where $tr\{\cdot\}$ is the trace of a matrix, $\Re\{\cdot\}$ is the real part of a complex number and $\frac{\partial(\cdot)}{\partial \chi_m}$ denotes the derivative with respect to mth element of χ.

While in [16] the CRLB for the Ricean source bearing estimation problem is studied in detail, one particularly informative characterization of the Ricean source bearing estimation problem and its relationship to the LOS and NLOS problems is given in Fig. 1.

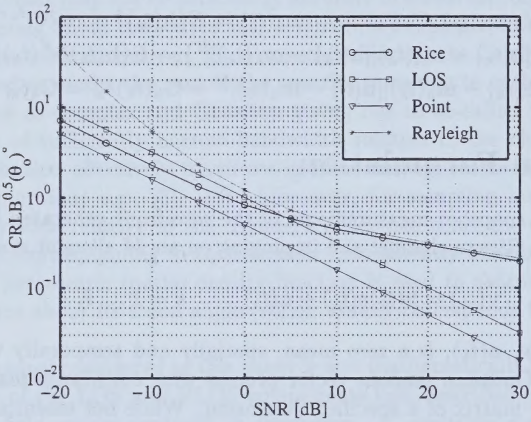


Figure 1: The CRLB on the bearing of a source in a Ricean channel with Rice factor (α^2/σ_b^2) of 1. (K=50 snapshots, M=6 sensors, $\theta_o = 30$ deg, $\Delta = 13$ deg). Also depicted are the bounds for the corresponding point source (Rice factor= ∞) and for a source in a Rayleigh channel (Rice factor = 0). The "LOS" bound is the bound for the LOS component only in the Ricean channel.

It shows $CRLB(\theta_0)$ as a function of signal to noise ratio (SNR). This bound is compared with the CRLB for a point source of the same SNR, and with the CRLB for the bearing of a source propagating in a NLOS Rayleigh channel. As intuitively expected, the performance of bearing estimation in a Ricean channel is bounded from below by the performance for a point source and bounded from above by the performance in a Rayleigh channel with the same SNR. Moreover, in a Ricean channel the estimation error does not reduce to zero with increasing SNR because of the bearing uncertainty in NLOS component.

Also, the CRLB for a point source of a reduced SNR, which represents the LOS component only, is depicted. The comparison between this LOS bound and the Rice bound aids in understanding the role of the NLOS component for bearing estimation in Ricean channel: at low SNR, where the estimation error is dominated by the noise, the additional power in the NLOS helps, so the CRLB is lower than in the case where the NLOS component is absent. However, at higher SNR, the NLOS component actually may be viewed as interference, and one can get better (even significantly better) bearing estimation if the NLOS component were absent.

5 Estimators

Our parameter of interest is the bearing of the source, θ_0. The remaining six unknown parameters are nuisance parameters. In [16] full and "compressed" maximum likelihood estimators for this problem are presented involving searches over seven and four dimensional spaces, respectively. The excessive complexity of these procedures motivates consideration of sub-optimal procedures which by-pass the need to estimate some or all of the nuisance parameters.

We consider two families of sub-optimal estimators. The first requires a two-dimensional search (over the bearing and the spreading parameters) and the second searches only over the bearing.

5.1 The Weighted Least Squares (WLS)

The WLS criteria [11] is given by:

$$l_{WLS}(\boldsymbol{\psi}) = tr\left[\mathbf{W}\left(\widehat{\mathbf{R}}_y - \mathbf{R}(\boldsymbol{\psi})\right)\mathbf{W}\left(\widehat{\mathbf{R}}_y - \mathbf{R}(\boldsymbol{\psi})\right)^H\right] \tag{7}$$

where $\mathbf{W}$ is a positive definite weighting matrix, $\widehat{\mathbf{R}}_y = \frac{1}{K}\sum_{k=1}^{K}\mathbf{y}(t_k)\mathbf{y}^H(t_k)$ is the sample data correlation matrix and $\mathbf{R}(\boldsymbol{\psi}) = \alpha^2\mathbf{a}(\theta_o)\mathbf{a}^H(\theta_o) + \sigma_b^2\mathbf{R}_b(\theta_o, \Delta) + \sigma_n^2\mathbf{I}_M$ is the true correlation matrix dependent on part of the unknown parameters vector $\boldsymbol{\psi} = [\theta_o, \Delta, \sigma_b^2, \sigma_n^2, \alpha^2]^T$.

The WLS estimate is obtained by minimizing the cost function $l_{WLS}(\boldsymbol{\psi})$ (7) with respect to the unknown parameter vector $\boldsymbol{\psi}$. The choice of $\mathbf{W}$ is usually such that the error covariance of $\boldsymbol{\psi}$ is minimized. However, in cases where the data vector is non-zero mean, the optimal choice if $\mathbf{W}$ cannot be easily found [11]. In [11] it is suggested to model the data as zero mean to facilitate calculation of $\mathbf{W}$. In this case, the optimum weighting matrix is $\mathbf{W} = \mathbf{R}^{-1}$. In practice, since $\mathbf{R}$ is unknown it is replaced by its consistent estimate $\widehat{\mathbf{R}}_y$, such that $\mathbf{W} = \widehat{\mathbf{R}}_y^{-1}$. This approach is applied to the (non-zero mean) Ricean channel model described in this paper and is therefore sub-optimal due to the modeling error. However, it is expected that the relative degradation in performance will not be excessive. In any case, the WLS criterion can now be written as:

$$l_{WLS}(\boldsymbol{\psi}) = tr\left[\left(\mathbf{R}\widehat{\mathbf{R}}_y^{-1} - \mathbf{I}\right)^2\right] \tag{8}$$

The resulting cost function is quadratic in $\alpha^2, \sigma_b^2, \sigma_n^2$ whose estimates can be shown to be given by the solution to:

$$\begin{bmatrix} \mathbf{a}^H\widehat{\mathbf{R}}_y^{-1}\mathbf{a}\mathbf{a}^H\widehat{\mathbf{R}}_y^{-1}\mathbf{a} & \mathbf{a}^H\widehat{\mathbf{R}}_y^{-1}\mathbf{R}_b\widehat{\mathbf{R}}_y^{-1}\mathbf{a} & \mathbf{a}^H\widehat{\mathbf{R}}_y^{-1}\widehat{\mathbf{R}}_y^{-1}\mathbf{a} \\ \mathbf{a}^H\widehat{\mathbf{R}}_y^{-1}\mathbf{R}_b\widehat{\mathbf{R}}_y^{-1}\mathbf{a} & tr\{\mathbf{R}_b\widehat{\mathbf{R}}_y^{-1}\mathbf{R}_b\widehat{\mathbf{R}}_y^{-1}\} & tr\{\widehat{\mathbf{R}}_y^{-1}\mathbf{R}_b\widehat{\mathbf{R}}_y^{-1}\} \\ \mathbf{a}^H\widehat{\mathbf{R}}_y^{-1}\widehat{\mathbf{R}}_y^{-1}\mathbf{a} & tr\{\mathbf{R}_b\widehat{\mathbf{R}}_y^{-1}\widehat{\mathbf{R}}_y^{-1}\} & tr\{\widehat{\mathbf{R}}_y^{-2}\} \end{bmatrix} \begin{bmatrix} \widehat{\alpha^2} \\ \widehat{\sigma_b^2} \\ \widehat{\sigma_n^2} \end{bmatrix} = \begin{bmatrix} \mathbf{a}^H\widehat{\mathbf{R}}_y^{-1}\mathbf{a} \\ tr\{\mathbf{R}_b\widehat{\mathbf{R}}_y^{-1}\} \\ tr\{\widehat{\mathbf{R}}_y^{-1}\} \end{bmatrix} \tag{9}$$

The WLS cost function (8) can be recast as a two dimensional search:

$$[\theta_o, \Delta] = \arg\min_{\theta,\Delta} tr\left\{\left[\left(\widehat{\alpha^2}\mathbf{a}\mathbf{a}^H(\theta_o) - \widehat{\sigma_b^2}\mathbf{R}_b(\theta_o,\Delta) - \widehat{\sigma_n^2}\mathbf{I}_M\right)\widehat{\mathbf{R}}_y^{-1} - \mathbf{I}\right]^2\right\}. \tag{10}$$

5.2 The Least Squares (LS)

The LS criteria [11] is given by:

$$l_{LS}(\psi) = tr\left[\left(\widehat{\mathbf{R}}_y - \mathbf{R}\right)\left(\widehat{\mathbf{R}}_y - \mathbf{R}\right)^H\right]. \tag{11}$$

Obviously, the LS is a special case of the WLS with $\mathbf{W} = \mathbf{I}$. The LS estimator is obtained by minimizing the cost function $l_{LS}(\psi)$, with respect to the unknown parameters vector ψ. Again, since the cost function is quadratic in $\alpha^2, \sigma_b^2, \sigma_n^2$, closed form expressions for their estimates can be obtained:

$$\begin{aligned}
\widehat{\sigma_n^2} &= \frac{[\beta M^2 - \gamma^2][tr\{\widehat{\mathbf{R}}_y\}M - \mathbf{a}^H\widehat{\mathbf{R}}_y\mathbf{a}] - [M^2 - \gamma][tr\{\widehat{\mathbf{R}_y}\mathbf{R}_b\}M^2 - \gamma\mathbf{a}^H\widehat{\mathbf{R}}_y\mathbf{a}]}{[\beta M^2 - \gamma^2][M-1]M - [M^2 - \gamma]M} \\
\widehat{\sigma_b^2} &= \frac{M^2 tr\{\widehat{\mathbf{R}}_y\mathbf{R}_b\} - \gamma\mathbf{a}^H\widehat{\mathbf{R}}_y\mathbf{a} - \widehat{\sigma_n^2}M[M^2 - \gamma]}{\beta M^2 - \gamma^2} \\
\widehat{\alpha^2} &= \frac{\mathbf{a}^H\widehat{\mathbf{R}}_y\mathbf{a} - M\widehat{\sigma_n^2} - \gamma\widehat{\sigma_b^2}}{M^2}, \qquad \beta = tr\left\{\mathbf{R}_b\mathbf{R}_b^H\right\} \quad \gamma = \mathbf{a}^H\mathbf{R}_b\mathbf{a}
\end{aligned} \tag{12}$$

The LS estimates can now be derived as:

$$[\theta_o, \Delta] = \arg\min_{\theta,\Delta} tr\left[\left(\widehat{\mathbf{R}}_y - \widehat{\alpha^2}\mathbf{a}(\theta_o)\mathbf{a}^H(\theta_o) - \widehat{\sigma_b^2}\mathbf{R}_b(\theta_o,\Delta) - \widehat{\sigma_n^2}\mathbf{I}_M\right)^2\right] \tag{13}$$

Both the WLS and the LS call for a two-dimensional search. Next, bearing estimators requiring only a one-dimensional search are considered.

5.3 The MVDR BF

The minimum variance distortionless response (MVDR) BF is a well known and popular beamforming technique for waveform and DOA estimation, e.g., [2]. The MVDR BF is designed to minimize the array output power subject to a distortionless response in direction θ. Fig. 1 along with the interference suppression capabilities of the MVDR BF motivate its use in estimating the bearing of the LOS component while suppressing the NLOS component of the received signal as interference.

The bearing estimator based on the MVDR BF is given by the bearing associated with the maximum of the MVDR spatial spectrum estimate:

$$[\widehat{\theta}_o]_{MV} = \max_{\theta} \frac{1}{\mathbf{a}^H(\theta)\widehat{\mathbf{R}}_y^{-1}\mathbf{a}(\theta)} \tag{14}$$

This procedure requires matrix inversion and a one dimensional search, which is computationally less expensive than the WLS and LS estimators.

5.4 The Conventional BF

The conventional ("delay and sum") BF for a bearing estimation, which is known to be the optimum MLE for bearing estimation of a single far field, point source in white noise [1]. In [17] it was shown that the BF is useful for bearing estimation of a distributed source. It is therefore believed to be especially attractive and robust for bearing estimation in a Ricean channel, where an LOS component exists.

The estimator is given by:

$$\widehat{\theta}_{BF} = \max_{\theta} \left\{ \frac{\mathbf{a}^H(\theta)\widehat{\mathbf{R}}_y\mathbf{a}(\theta)}{\mathbf{a}^H(\theta)\mathbf{a}(\theta)} \right\}.$$

The conventional BF is simpler than the MVDR BF since no matrix inversion is required.

6 Simulation Study

This section presents a simulation study for the performance of the methods considered in the previous section for estimating the bearing of a source propagating in a Ricean channel. Consider the following nominal scenario: $K = 50$ snapshots are taken using a uniform, linear array of $M = 6$ sensors spaced at half of a wavelength. The LOS component is subject to a Doppler shift of approximately 0.15 times the snapshot sampling frequency while the NLOS component is assumed to be spatially distributed about the mean angle $\theta_o = 30°$ and uniform spreading about the mean angle with $\Delta = 13°$ (for a total of $2\Delta = 26°$ of spreading). The nominal SNR is set to 10dB and nominal Rice factor is one. The transmitted signal is of constant unit modulus. The following figures presents simulation results for the root mean squared error (RMSE) of the bearing estimates as a function of variations of the following parameters about the nominal scenario: the angular spreading of the NLOS component, Δ, the SNR, the Rice factor α^2/σ_b^2, the number of snapshots, K.

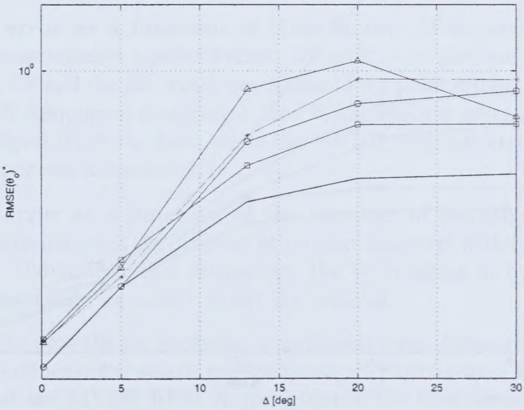


Figure 2: The RMSE of the bearing vs. spreading (Δ) for the different estimators relative to the CRLB (solid line) WLS - squares, LS - circles, MVDR - triangles, Conv. BF - ast.

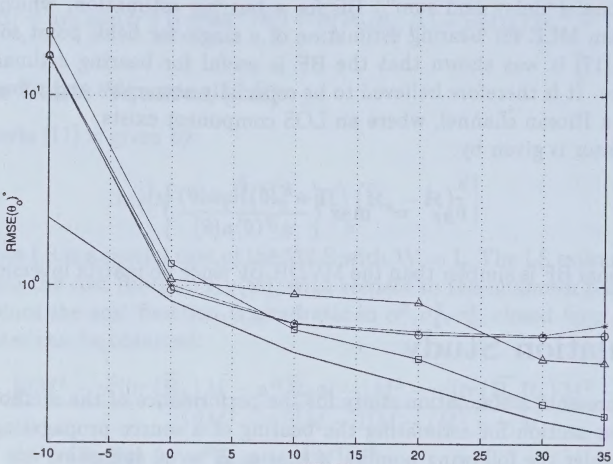


Figure 3: The RMSE of the bearing vs. SNR, CRLB (solid line), WLS - squares, LS - circles, MVDR - triangles, Conv. BF - ast.

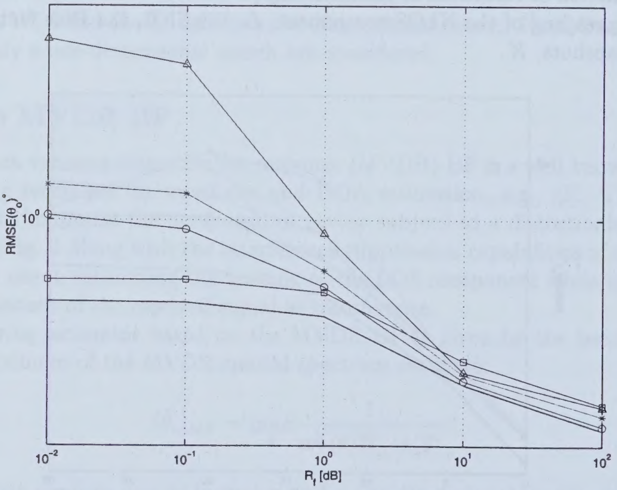


Figure 4: The RMSE of the bearing vs. Rice factor (R_f), CRLB (solid line), WLS - squares, LS - circles, MVDR - triangles, Conv. BF - ast.

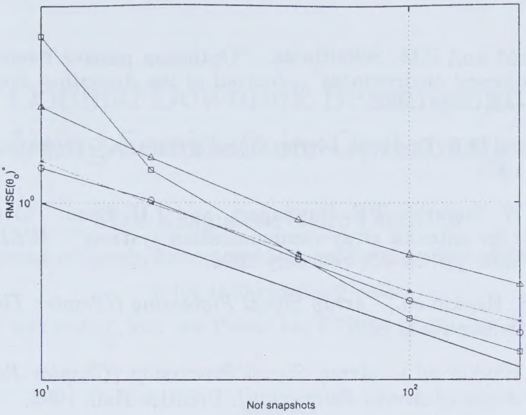


Figure 5: The RMSE of the bearing vs. number of snapshots, CRLB (solid line), WLS - squares, LS - circles, MVDR - triangles, Conv. BF - ast.

6.1 Discussion of the results

- **Estimation error as a function of spreading**. As in the CRLB, most estimation errors increase with spreading. Note, however, that this is not the case with the MVDR. Its ability to null the NLOS component increases with the spreading, and therefore at large spreading its bearing estimation error decreases.
- **Estimation error as a function of SNR**. As in the CRLB, the performance of the different estimators improves with increasing SNR. It is interesting to note that at high SNR the performance of the LS and the conventional BF estimators are more affected by the NLOS component of the signal.
- **Estimation error as a function of Rice factor**. With large Rice factor the Ricean source approaches a point source. All estimators perform well in this case, especially the BF and the LS, which are optimal for a point source. With small Rice factor the LOS component disappears. The WLS, which is asymptotically optimal for a NLOS signal [8], is the best, while the MVDR, which is highly dependent on the LOS component, is the worst.
- **Estimation error as a function of the number of snapshots**. As with the CRLB, the performance of the different estimators improves with increasing number of snapshots. Unlike the other estimators, the WLS seems to be almost efficient even though its weighting matrix is not the optimal.

It is interesting to note the no estimator is uniformly best. Moreover, the estimators which require a two-dimensional search are not necessarily better than the (significantly) simpler conventional and MVDR BF's. In particular, if the Rice factor is not too small and/or the spreading is not too large, the performance of the conventional BF is close to optimal. If necessary, when these conditions are not satisfied, the more complex LS/WLS procedures may be applied to obtain performance close to the CRLB.

Also, in [16] an indicator which can be used as a quality factor for the conventional BF bearing estimate is developed.

References

[1] V.H. MacDonald and P.M. Schultheiss. "Optimum passive bearing estimation in a spatially incoherent environment". *Journal of the Acoustical Society of America*, vol. 46: pp. 37–43, July 1969.

[2] D.H. Johnson and D.E. Dudgeon. *Array Signal Processing–Techniques and Concepts.* Prentice Hall, 1993.

[3] R.B. Ertel, K.W. Sowerby, T.S. Rappaport, and J.H. Reed. "Overview of spatial channel models for antenna array communication systems". *IEEE Personal Communications*, pages pp. 10–22, February 1998.

[4] N.L. Owsley (S. Haykin ed.). *Array Signal Processing (Chapter Three–Sonar Array Processing).* Prentice-Hall, 1985.

[5] S. Haykin (S. Haykin ed.). *Array Signal Processing (Chapter Four–Radar Array Processing for Angle of Arriva Estimation).* Prentice-Hall, 1985.

[6] S. Valaee, B. Champagne, and P. Kabal. "Parametric localization of distributed sources". *IEEE Trans. Sig. Proc.*, vol. 43: pp. 2144–2153, September 1995.

[7] K.M.Wong Y. Meng, P. Stoica. "Estimation of the directions of arrival of spatially dispersed signals in array processing". *IEE Proceedings*, 1996.

[8] T. Trump and B. Ottersten. "Estimation of nominal direction of arrival and angular spread using an array of sensors". *Signal Processing*, vol. 50: pp. 57–69, April 1996.

[9] J. Goldberg and H. Messer. "Inherent limitations in the localization of a coherently scattered source". *IEEE Trans. Sig. Proc.*, vol. 46: pp. 3441–3444, December 1998.

[10] R. Raich, J. Goldberg, and H. Messer. "Bearing estimation for a distributed source: Modeling, inherent accuracy limitations, and algorithms". to appear in *IEEE Trans. on Signal Processing.*

[11] K.V.S. Hari and B. Ottersten. "Parameter estimation using a sensor array in a Ricean fading channel". In *Proceedings of SPCOM'97*, 1997.

[12] T.T. Tjhung, X.D. Dong, F. Adachi, and K.H. Tan. "On diversity reception of narrowband 16 STAR-QAM" in fast Rician fading". *IEEE Transactions on Vehicular Technology*, vol. 46: pp. 923–932, November 1997.

[13] K.I. Pedersen, P.E. Mogensen, B.H. Fleury, F. Frederiksen, K. Olesen, and S.L. Larsen. "Analysis for time, azimuth and Doppler dispersion in outdoor radio channels". In *Proceedings of ACTS Summit*, Aalborg, Denmark, 1997.

[14] P. Zetterberg. *Mobile Cellular Communications with Base Station Antenna Arrays: Spectrum Efficiency Algorithms, and Propagation Models.* PhD thesis, Royal Institute of Technology, Stockholm, Sweden, 1997.

[15] S.M. Kay. *Fundamentals of Statistical Signal Processing - Estimation Theory.* Prentice Hall, 1993.

[16] G. Fuks. "Source parameter estimation for a Ricean spatio-temporal channel". Master's thesis, Tel-Aviv University, 1999.

[17] R. Raich. "Direction of arrival estimation for distributed sources-modeling,bounds, and estimators". Master's thesis, Tel-Aviv University, 1998.

Optimal Downlink Beamforming Using Semidefinite Optimization

Mats Bengtsson, Björn Ottersten
Department of Signals, Sensors, and Systems, Royal Institute of Technology
S-100 44 Stockholm, Sweden
Email: `matsb@s3.kth.se`, Phone: +46 8 790 84 63, Fax: +46 8 790 73 29

Abstract

When using antenna arrays at the base station of a cellular system, one critical aspect is the transmission strategy. An optimal choice of beamformers for simultaneous transmission to several co-channel users must be solved jointly for all users and base stations in an area. We formulate an optimal transmit strategy and show how the solution can be calculated efficiently using interior point methods for semidefinite optimization. The algorithm minimizes the total transmitted power under certain constraints to guarantee a specific quality of service. The method provides large flexibility in the choice of constraints and can be extended to be robust to channel perturbations.

1 Introduction

The use of antenna arrays brings new possibilities in the design of mobile communications systems. It is well known that the system capacity is more limited in the downlink than in the uplink [7, 22]. Yet, the literature on beamforming for transmission is relatively small, compared to the well investigated topic of beamforming for a receiving antenna array. Some results on downlink beamforming can be found in [8, 11, 12, 15, 21, 22].

The simplest transmit strategy is to use standard beamforming, i.e., to point the main lobe of the antenna array in the direction of the specific receiver [13]. Knowing the channel to nearby co-channel users, it is possible to actively suppress the signal to the interfered users. In [15], Rashid-Farrokhi et al. formulate the beamforming design as a constrained optimization problem and present an algorithm that is shown to find the global optimum.

In this paper, we consider the same optimization problem, namely to minimize the total transmitted power while maintaining a certain quality of service for all users, but present an alternative solution using convex optimization. This gives several advantages. First of all, the solution can be efficiently calculated using standard algorithms for semidefinite optimization [16, 19] with guaranteed convergence speed. Secondly, with this technique, it is very easy to introduce modifications, for example adding extra constraints on the dynamic range or adding increased robustness to channel estimation errors. In most cases our method gives a solution in the form of a standard fixed beamformer, in the remaining cases the solution can be implemented using a time-varying beamformer – a space time code [17]. Note however that the problem formulations we consider here will not necessarily give any coding gain.

In [20], Visotsky and Madhow consider the algorithm in [15] for the special case of rank one channels and claim that it does not give the optimal solution unless an extra scaling of the problem is introduced. However, our results confirm that the algorithm in [15] does indeed give the correct solution also without any rescaling. Furthermore, we show that for rank one channels, the problem is essentially constrained by a convex second-order cone [10], Thus, any reasonable iterative minimization procedure should converge to the global optimum.

To our knowledge, convex optimization for beamformer design has previously only been used with traditional filter design-type constraints on pass bands and side-lobe levels [9], but not within the framework of statistically optimal beamforming.

When the antenna array is used as a receiver, i.e. in uplink mode, the instantaneous channel can be estimated directly from the received data, whereas in the downlink, the transmitting beamformer must be based on information collected in the uplink. Several schemes have been proposed for the transformation from uplink to downlink. In a Time Division Duplex (TDD) system with sufficiently short time slots, the downlink channel is virtually identical to the uplink channel, whereas in a Frequency Division Duplex (FDD) system, the channel fades independently at the two duplex frequencies. However, a statistical model of the downlink channel can be obtained from the collected uplink data using a physical model [4, 23] or model-free techniques [3, 8]. Throughout this paper, we assume that a stochastic characterization of the downlink transmission channel is known at the base station for all co-channel mobiles within its range (possible located in neighboring cells).

We study the joint design of the beamformers for all co-channel users within a large region which could include several cells. The optimal solution must be solved jointly since every transmitted signal will affect all receivers. In a practical system, a decentralized suboptimal solution may be preferable to reduce the overhead of collecting all channel measurements. The optimal solution does still provide a valuable benchmark for evaluation of other algorithms and for use in system simulations and capacity studies.

The paper is organized as follows. The system model and the basic assumptions are presented in Section 2. In Section 3, we first state the problem and show how it can be solved using a semidefinite relaxation. Then we show how to incorporate additional constraints and give a short overview of some suboptimal solutions. Finally we present a few numerical examples in Section 4.

2 System Model

We consider a system where a number of co-channel mobile users are served by one or more base stations and each base station is equipped with an antenna array. We will design the beamformers assuming a stationary scenario where the fast (Rayleigh) fading is described by its second order properties. We also assume narrow-band signals without any time dispersion, i.e., the channel fading is frequency flat. The model can easily be extended to frequency selective channels, taking both co-channel interference and inter-symbol interference into account, see [15].

In the baseband, the signal received by the ith mobile, $r_i(t)$, is given by

$$r_i(t) = \sum_{k=1}^{K} \mathbf{v}_{i,k}^* \mathbf{x}_k(t) + n_i(t) , \tag{1}$$

where $(\cdot)^*$ denotes Hermitian vector transpose. Here, $\mathbf{x}_k(t)$ is the complex valued $m \times 1$ vector of the baseband signals transmitted at the antenna elements of base station k, $n_i(t)$ is zero mean white complex noise with variance σ_i^2. The channel from base station k to mobile i is given by the random complex valued vector $\mathbf{v}_{i,k}$ with correlation matrix

$$\mathbf{R}_{i,k} = \mathrm{E}[\mathbf{v}_{i,k}\mathbf{v}_{i,k}^*] \,. \tag{2}$$

In the special case of line of sight transmission, $\mathbf{R}_{i,k} = \mathbf{h}_{i,k}\mathbf{h}_{i,k}^*$, where $\mathbf{h}_{i,k}$ is a deterministic array response vector, but in general, $\mathbf{R}_{i,k}$ can have any rank because of specular or diffuse multi-path propagation.

Each mobile is allocated to one base station and $\kappa(i)$ is used to denote the base station allocated for mobile i. Likewise, $\mathcal{I}(k) = \{i|\kappa(i) = k\}$ denotes the indices of the set of mobiles allocated to base k. We will use the shorthand notation $\mathbf{R}_i$ for $\mathbf{R}_{i,\kappa(i)}$.

The signal transmitted at base k is given by

$$\mathbf{x}_k(t) = \sum_{i\in\mathcal{I}(k)} \mathbf{w}_i s_i(t) \,, \tag{3}$$

where $s_i(t)$ is the scalar data sequence intended for user i and $\mathbf{w}_i$ is the beamforming weight vector for transmission from base $\kappa(i)$ to mobile i. For simplicity we assume that all $s_i(t)$ are uncorrelated and have the same power $\mathrm{E}[|s_i(t)|^2] = 1$.

3 Algorithms

We consider the design of the beamformers $\mathbf{w}_i$ given estimates of all $\mathbf{R}_{i,k}$ and σ_i^2. The goal is to minimize to total transmitted power

$$P = \sum_{i=1}^{d} \|\mathbf{w}_i\|^2 \tag{4}$$

while maintaining an acceptable quality of service for all users. We will first consider an SINR threshold and then show how additional constraints can be used to gain increased robustness to estimation errors in the channel covariance matrices or to handle a limited dynamic range.

3.1 Optimal Beamforming

We consider the same problem formulation as in [15, 20], namely to minimize the total transmitted power under the constraint that the received SINR at each mobile is above a certain threshold, $\mathrm{SINR}_i \geq \gamma_i$. This gives the following optimization problem

$$\begin{aligned} &\min \sum_{i=1}^{d} \|\mathbf{w}_i\|^2 \\ &\text{s.t.} \frac{\mathbf{w}_i^*\mathbf{R}_i\mathbf{w}_i}{\sum_{n\neq i} \mathbf{w}_n^*\mathbf{R}_{i,\kappa(n)}\mathbf{w}_n + \sigma_i^2} \geq \gamma_i,\ i = 1,\ldots,d \,, \end{aligned} \tag{5}$$

or equivalently,

$$\begin{aligned} &\min \sum_{i=1}^{d} \mathbf{w}_i^*\mathbf{w}_i \\ &\text{s.t. } \mathbf{w}_i^*\mathbf{R}_i\mathbf{w}_i - \gamma_i \sum_{n\neq i} \mathbf{w}_n^*\mathbf{R}_{i,\kappa(n)}\mathbf{w}_n \geq \gamma_i\sigma_i^2,\ i = 1,\ldots,d \,. \end{aligned} \tag{6}$$

It is easy to show that all constraints must be active at the optimum [20], thus the inequality in (6) can be replaced by an equality.

For the special case of rank one channels, $\mathbf{w}_n^* \mathbf{R}_{i,\kappa(n)} \mathbf{w}_n = |\mathbf{w}_m^* \mathbf{h}_{i,\kappa(m)}|^2$ and we can without loss of generality add the constraints $\mathbf{w}_i^* \mathbf{h}_i \geq 0$ which gives the following problem.

$$\begin{aligned} &\min \sum_{i=1}^{d} \mathbf{w}_i^* \mathbf{w}_i \\ &\text{s.t. } (\mathbf{w}_i^* \mathbf{h}_i)^2 \geq \gamma_i \left(\sum_{n \neq i} \mathbf{w}_n^* \mathbf{R}_{i,\kappa(n)} \mathbf{w}_n + \sigma_i^2 \right), \\ &\qquad \mathbf{w}_i^* \mathbf{h}_i \geq 0,\ i = 1, \ldots, d\,. \end{aligned} \tag{7}$$

However, these constraints are just an affine transformation of the convex second-order cone $\{\mathbf{x}, y |\ \|\mathbf{x}\|^2 \leq y^2, y \geq 0\}$, which shows that this is a convex problem which can be efficiently solved using standard methods for convex optimization [10].

In the general case, the original constraint set, (6) is not convex, but we will show that the problem still can be efficiently solved using convex optimization.

To this end, introduce the matrices $\mathbf{W}_i = \mathbf{w}_i \mathbf{w}_i^*$ and use the rule $\mathbf{w}^* \mathbf{R} \mathbf{w} = \mathrm{Tr}[\mathbf{R} \mathbf{w} \mathbf{w}^*] = \mathrm{Tr}[\mathbf{R} \mathbf{W}]$ to rewrite the problem into

$$\begin{aligned} &\min \sum_{i=1}^{d} \mathrm{Tr}[\mathbf{W}_i] \\ &\text{s.t. } \mathrm{Tr}[\mathbf{R}_i \mathbf{W}_i] - \gamma_i \sum_{n \neq i} \mathrm{Tr}[\mathbf{R}_{i,\kappa(n)} \mathbf{W}_n] = \gamma_i \sigma_i^2, \\ &\qquad \mathbf{W}_i = \mathbf{W}_i^*, \\ &\qquad \mathbf{W}_i \succeq 0,\ i = 1, \ldots, d\,. \end{aligned} \tag{8}$$

Here, the notation $\mathbf{W} \succeq 0$ means that $\mathbf{W}$ is positive semidefinite. Note that with the additional constraints $\mathrm{rank}[\mathbf{W}_i] = 1$, (8) is equivalent to (6), thus if the solution of (8) has $\mathrm{rank}[\mathbf{W}_i] = 1$ for all i, then it is also a solution of (5). Relaxing the rank of $\mathbf{W}_i$ gives a semidefinite optimization problem with a solution that always is a lower bound for the original problem. This technique is called a Lagrangian relaxation [2, 19] since it is the Lagrangian dual of the dual of the original problem. For this specific problem, however, we can show a much stronger result, namely.

Theorem 3.1. *If (8) is feasible, than it has at least one solution where* $\mathrm{rank}[\mathbf{W}_i] = 1$, *for all* $i = 1, \ldots, d$.

Proof. See [5]. □

If the relaxation has several minima, there is no guarantee that all solutions have rank one. One counterexample is given by $\sigma_i^2 = 1$, $\gamma_i = 1/2$, $\kappa(i) = 1$ and

$$\mathbf{R}_{11} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad \mathbf{R}_{21} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad \mathbf{R}_{31} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix},$$

which has several solutions including

$$\mathbf{W}_1 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & \frac{4}{3} & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad \mathbf{W}_2 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & \frac{7}{6} & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad \mathbf{W}_3 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \frac{1}{2} \end{bmatrix}$$

but also

$$\mathbf{W}_1 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & \frac{2}{3} & 0 \\ 0 & 0 & \frac{2}{3} \end{bmatrix} \quad \mathbf{W}_2 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & \frac{5}{6} & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad \mathbf{W}_3 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \frac{5}{6} \end{bmatrix} .$$

Thus, there is no guarantee that an algorithm for solving semidefinite problems will give the desired rank one solution. However, in practice adding a small perturbation to one of the $\mathbf{R}_i$ matrices will change the problem into having only a rank one solution. An alternative approach to construct a rank one solution from any given solution is given in [5].

3.2 Additional Constraints

With the relaxation technique, it is easy to change the constraints or add more constraints. To illustrate the principle, we will show how to add constraints on the dynamic range of the power amplifier at each antenna element. The total transmitted power of element l of array k is

$$\mathrm{E}[|[\mathbf{x}_k]_l|^2] = \sum_{i \in \mathcal{I}(k)} |[\mathbf{w}_i]_l|^2 = \sum_{i \in \mathcal{I}(k)} [\mathbf{W}_i]_{ll}$$

Thus upper and lower bounds μ_{kl}^U and μ_{kl}^L, respectively, on the average power of an antenna element can be formulated by constraints of the form

$$\mu_{kl}^L \leq \sum_{i \in \mathcal{I}(k)} [\mathbf{W}_i]_{ll} \leq \mu_{kl}^U .$$

Similarly, we could set limits on the relative dynamic range of a single element in comparison to the total power for the whole array by replacing μ_{kl} with $\mu_{kl} \sum_{i \in \mathcal{I}(k)} \mathrm{Tr}[\mathbf{W}_i]$.

Since all these constraints are linear in the elements of $\mathbf{W}_i$, the resulting problem will still be semidefinite. However, we can not in general expect the solutions to have rank one. For system evaluations and simulations, the solution provides a lower bound for the problem, but a high rank solution can actually be implemented also in a practical system. If we allow for time varying beamformers $\mathbf{w}_i(t)$, $\mathbf{W}_i$ can be interpreted as the covariance matrix of the beamformer, $\mathbf{W}_i = \mathrm{E}[\mathbf{w}_i \mathbf{w}_i^*]$. Thus, one possible implementation is to use a random sequence of vectors with covariance $\mathbf{W}_i$, as a time varying beamformer. Compare to [14] where a similar interpretation of $\mathbf{W}_i$ is used to evaluate the usefulness of space-time coding for different scenarios.

3.3 Increased Robustness

A common problem in connection with optimal uplink beamforming is signal cancellation caused by estimation errors in the channel covariances [6]. We could expect similar problems in the downlink, so it is interesting to introduce a design strategy that is robust to small errors in $\mathbf{R}_{ik}$. Assume that the true channel has a covariance matrix $\hat{\mathbf{R}}_{ik} + \Delta_{ik}$, where $\hat{\mathbf{R}}_{ik}$ is the available estimate. The goal is to find $\mathbf{w}_i$ such that the SINR constraints in (5) hold for all choices of Δ_{ik} with $\|\Delta_{ik}\| \leq \epsilon_{ik}$ for some matrix norm $\|\cdot\|$. Since

$$\max_{\|\Delta\| \leq \epsilon} |\mathrm{Tr}[\mathbf{X}\Delta]| = \epsilon \|\mathbf{X}\|_* ,$$

where $\|\cdot\|_*$ is the dual norm of $\|\cdot\|$. The resulting problem can be formulated as

$$\begin{aligned} &\min \sum_{i=1}^{d} \mathrm{Tr}[\mathbf{W}_i] \\ &\text{s.t. } \mathrm{Tr}[\hat{\mathbf{R}}_i \mathbf{W}_i] - \gamma_i \sum_{n \neq i} \mathrm{Tr}[\hat{\mathbf{R}}_{i,\kappa(n)} \mathbf{W}_n] \geq \gamma_i \sigma_i^2 + \epsilon_i \|\mathbf{W}_i\|_* + \gamma_i \sum_{n \neq i} \epsilon_{i,\kappa(n)} \|\mathbf{W}_n\|_*, \\ &\qquad \mathbf{W}_i = \mathbf{W}_i^*, \\ &\qquad \mathbf{W}_i \succeq 0,\ i = 1, \dots, d\,. \end{aligned} \tag{9}$$

If we use the spectral norm of Δ_{ik}, i.e. put a limit of the maximum eigenvalue $\lambda_{\max}(\Delta_{ik}) \leq \epsilon_{ik}$, then the dual norm is the absolute sum of eigenvalues $\|\mathbf{W}\|_* = \sum |\lambda_k(\mathbf{W})|$ and since all $\mathbf{W}_i$ are positive semidefinite, the optimal robust design is solved by the following semidefinite problem

$$\begin{aligned} &\min \sum_{i=1}^{d} \mathrm{Tr}[\mathbf{W}_i] \\ &\text{s.t. } \mathrm{Tr}[\hat{\mathbf{R}}_i \mathbf{W}_i] - \gamma_i \sum_{n \neq i} \mathrm{Tr}[\hat{\mathbf{R}}_{i,\kappa(n)} \mathbf{W}_n] \geq \gamma_i \sigma_i^2 + \epsilon_i \,\mathrm{Tr}[\mathbf{W}_i] + \gamma_i \sum_{n \neq i} \epsilon_{i,\kappa(n)} \,\mathrm{Tr}[\mathbf{W}_n], \\ &\qquad \mathbf{W}_i = \mathbf{W}_i^*, \\ &\qquad \mathbf{W}_i \succeq 0,\ i = 1, \dots, d\,. \end{aligned} \tag{10}$$

3.4 Suboptimal Solutions

Even if an efficient algorithm is available for the joint design of all beamformers, a decentralized algorithm could be preferable in a running network to decrease the overhead communication between different base stations.

A heuristic approach is to determine each beamformer separately, keeping the received SNR at the mobile of interest above the threshold μ_i and the total transmitted power to the interfered users below the threshold ξ_i,

$$\mathbf{w}_i^* \Big(\sum_{n \neq i} \mathbf{R}_{n,\kappa(i)} \Big) \mathbf{w}_i \triangleq \mathbf{w}_i^* \mathbf{Q}_i \mathbf{w}_i \leq \xi_i\,.$$

Minimizing the transmitted power gives the following optimization problem,

$$\begin{aligned} \mathbf{w}_i = \arg\min P_i &= \|\mathbf{w}_i\|^2 \\ \text{s.t. } \mathbf{w}_i^* \mathbf{R}_i \mathbf{w}_i &\geq \mu_i \sigma_i^2 \\ \mathbf{w}_i^* \mathbf{Q}_i \mathbf{w}_i &\leq \xi_i\,. \end{aligned} \tag{11}$$

The same relaxation as in the previous sections gives the semidefinite optimization problem,

$$\begin{aligned} \mathbf{W}_i = \arg\min &\ \mathrm{Tr}[\mathbf{W}_i] \\ \text{s.t. } &\mathrm{Tr}[\mathbf{W}_i \mathbf{R}_i] \geq \mu_i \sigma_i^2 \\ &\mathrm{Tr}[\mathbf{W}_i \mathbf{Q}_i] \leq \xi_i \\ &\mathbf{W}_i^* = \mathbf{W}_i \\ &\mathbf{W}_i \succeq 0\,. \end{aligned}$$

Also for this problem it is possible to show that there always exists an optimal solution $\mathbf{W}_i$ of rank one.

A related strategy, which has been suggested in several references including [6, 8, 23], is to use

$$\mathbf{w}_i = \arg\max \frac{\mathbf{w}_i^* \mathbf{R}_i \mathbf{w}_i}{\mathbf{w}_i^* (\alpha \mathbf{I} + \sum_{n \neq k} \mathbf{R}_n) \mathbf{w}_i}, \tag{12}$$

where the optimum is given as the solution of a generalized eigenvalue problem. The parameter $\alpha \geq 0$ can be interpreted as a Lagrange multiplier of (11) and determines the trade-off between interference suppression and a low total transmission power [6].

Finally, we mention traditional beamforming,

$$\mathbf{w}_i = \arg\max \frac{\mathbf{w}_i^* \mathbf{R}_i \mathbf{w}_i}{\mathbf{w}_i^* \mathbf{w}_i}, \tag{13}$$

i.e., to use the principal eigenvector of $\mathbf{R}_i$.

4 Numerical Examples

We illustrate the performance of the suggested downlink algorithms in a simulated scenario with three users served by a single base station. One mobile is located at $\theta_1 = 10°$ relative array broadside and the two others at directions $\theta_{2,3} = 10° \pm \Delta$, where$\Delta$ is varied from 5°to 25°. The transmitting antenna array is linear and has $m = 8$ elements spaced half a wavelength. Each user is surrounded by a large number of local scatterers corresponding to a spread angle of $\sigma_\theta = 2°$, as seen from the base station. Thus, the channel covariance matrix is well approximated by [1, 18],

$$[\mathbf{R}(\theta, \sigma_\theta)]_{kl} = e^{j\pi(k-l)\sin\theta} e^{-\frac{(\pi(k-l)\sigma_\theta \cos\theta)^2}{2}}.$$

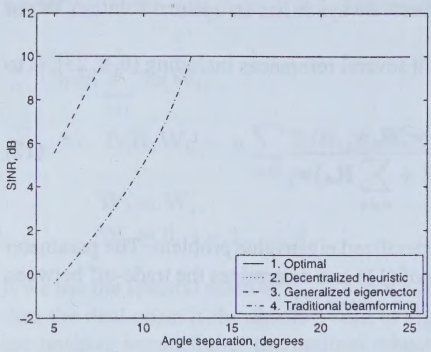
We compare the following four transmit strategies.

1. The joint optimal design according to Section 3.1.
2. The decentralized heuristic design given by (11).
3. The heuristic generalized eigenvector beamformer given by (12).
4. Traditional beamforming given by (13).

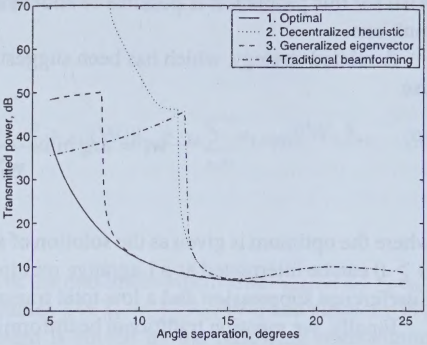
In all cases, the beamformers are scaled, if possible, such that $\text{SINR}_i = \gamma_i$ for all users. In the cases where no such scaling is possible, the scaling is chosen to maximize the worst SINR among the users, thus all users receive the same SINR in all the examples. The semidefinite problems were solved using a primal-dual interior point method [16].

The SINR threshold was set to $\gamma_i = 10$dB and in (11), the thresholds were set to $\mu_i \sigma_i^2 / xi_i = 10$dB. In (12), the parameter α was arbitrarily set to

$$\alpha = \frac{0.1}{m} \operatorname{Tr}\Big[\sum_{n \neq k}^{d} \mathbf{R}_n\Big], \tag{14}$$



1. Received signal to interference ratio at the middle user.



2. Total transmitted power, relative the noise level at each receiver.

10% of the gain level of the interfered channels.

Figure 1 shows the received SINR level for the users. The total transmitted power needed to achieve this level for all users is shown in Figure 2.

Because of the scaling used for the solutions, Figure 1 mainly shows when it is possible to achieve the desired SINR at all receivers using the different strategies. The decentralized solution in (11) did not produce any feasible solution when the source separation was $< 7°$.

As could be expected, the traditional beamformer performs worst, since it does not take the interfered users into account. However, when the users are sufficiently separated such that the traditional beamformer does give a feasible solution, then it coincides with the heuristic solution in (11), which thus can be seen as a generalization of the traditional beamforming. The performance of the generalized eigenvector solution in (12) depends on the choice of α, but in this example it performs almost as well as the optimal solution for the cases where it gives a feasible solution. In the difficult scenarios where the users are very closely separated, the cost, in terms of total transmitted power, is very large in order to obtain a feasible solution.

5 Conclusions

We have proposed strategies and algorithms for the design of beamformers that keep the total transmitted power at a minimum, still maintaining a certain level of quality. The technique can be applied both on systems with inter-cell channel reuse as well as systems with at most one co-channel user per cell.

The quality constraints are given in terms of the average signal to interference plus noise ratio received at each mobile. We have shown how the resulting optimization problem can be efficiently solved using standard tools for semidefinite optimization. This is a surprising result, since the original problem is non-linear and non-convex. It is also possible to show a similar result for the problem of joint power control and beamforming for the uplink.

Compared to the previously published algorithm [15] for the problem, the solution using convex optimization has guaranteed convergence speed and will quickly detect the infeasible situations where no solution can be found, but the main advantage is the flexibility offered in the choice of constraints. Also robust beamforming is easily incorporated. Even though the method will not always produce a normal time-invariant beamformer when extra constraints are added, the solution does still provide a useful benchmark for e.g. capacity studies.

Acknowledgment

The authors would like to express their gratitude to professor Stephen Boyd at the Information Systems Laboratory, Stanford University and to associate professor Anders Forsgren and his colleagues at the division of optimization and systems theory, Royal Institute of Technology, Stockholm for their valuable ideas and comments.

References

[1] F. Adachi, M.T. Feeney, A.G. Williamson, and J.D. Parsons. Cross correlation between the envelopes of 900 MHz signals received at a mobile radio base station site. *IEE Proceedings, Pt. F*, 133(6):506–512, October 1986.

[2] Kurt Anstreicher and Henry Wolkowicz. On Lagrangian relaxation of quadratic matrix constraints. Technical Report CORR 98-24, Department of Combinatorics and Optimization, University of Waterloo, Waterloo, Ontario, Canada, 1998. To appear in SIAM Journal on Matrix Analysis and Applications.

[3] T. Asté, P. Forster, L. Féty, and S. Mayrargue. Downlink beamforming avoiding DOA estimation for cellular mobile communications. In *Proc. IEEE ICASSP 98*, volume 6, pages 3313–3316, 1997.

[4] Mats Bengtsson and Björn Ottersten. On approximating a spatially scattered source with two point sources. In *Proc. NORSIG'98*, pages 45–48, June 1998.

[5] Mats Bengtsson and Björn Ottersten. Optimal transmit beamforming using convex optimization. In preparation, 1999.

[6] Henry Cox, Robert M. Zeskind, and Mark M. Owen. Robust adaptive beamforming. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 35(10):1365–1375, October 1987.

[7] Lal C. Godara. Applications of antenna arrays to mobile communications. I. Performance improvement, feasibility, and system considerations. *Proceedings of IEEE*, 85(7):1031–1060, July 1997.

[8] Jason Goldberg and Javier R. Fonollosa. Downlink beamforming for cellular mobile communications. In *Proceedings of VTC'97*, volume 2, pages 632–636, 1997.

[9] Hervé Lebret and Stephen Boyd. Antenna array pattern synthesis via convex optimization. *IEEE Transactions on Signal Processing*, 45(3):526–532, March 1997.

[10] Miguel Sousa Lobo, Lieven Vandenberghe, Stephen Boyd, and Hervé Lebret. Applications of second-order cone programming. *Linear Algebra Appl.*, 284(1-3):193–228, 1998.

[11] Hussein K. Mecklai and Rick S. Blum. Transmit antenna diversity for wireless communications. In *Proceedings IEEE International Conference on Communications 95*, volume 3, pages 1500 – 1504, 1995.

[12] Guiseppe Montalbano and Dirk T.M. Slock. Spatio-temporal array processing for matched filter bound optimization in SDMA downlink transmission. In *URSI Internat. Symp. Signals, Systems and Electronics*, pages 416 – 421, 1998.

[13] Robert A. Monzingo and Thomas W. Miller. *Introduction to Adaptive Arrays*. Wiley, 1980.

[14] Aradhana Narula, Michael J. Lopez, Mitchell D. Trott, and Gregory W. Wornell. Efficient use of side information in multiple-antenna data transmission over fading channels. *IEEE Journal on Selected Areas in Communications*, 16(8):1423–1436, October 1998.

[15] Farrokh Rashid-Farrokhi, K.J. Ray Liu, and Leandros Tassiulas. Transmit beamforming and power control for cellular wireless systems. *IEEE Journal on Selected Areas in Communications*, 16(8):1437–1450, October 1998.

[16] Jos F. Sturm. *Using SeDuMi 1.02, a MATLAB Toolbox for Optimization over Symmetric Cones*. Communications Research Laboratory, McMaster University, Hamilton, Canada, `http://www.unimaas.nl/~sturm/software/sedumi.html`, 1998.

[17] Vahid Tarokh, Nambi Seshadri, and A. R. Calderbank. Space-time codes for high data rate wireless communication: Performance criterion and code construction. *IEEE Transactions on Information Theory*, 44(2):744 – 765, March 1998.

[18] Tõnu Trump and Björn Ottersten. Estimation of nominal direction of arrival and angular spread using an array of sensors. *Signal Processing*, 50(1-2):57–69, April 1996.

[19] Lieven Vandenberghe and Stephen Boyd. Semidefinite programming. *SIAM Review*, 38(1):49–95, March 1996.

[20] Eugene Visotsky and Upamanyu Madhow. Optimum beamforming using transmit antenna arrays. In *Proceedings of VTC'99*, volume 1, pages 851–856, Houston, Texas, USA, May 1999. IEEE.

[21] Gregory W. Wornell and Mitchell D. Trott. Efficient signal processing techniques for exploiting transmit antenna diversity on fading channels. *IEEE Transactions on Signal Processing*, 45(1):191–204, January 1997.

[22] Per Zetterberg. *Mobile Cellular Communications with Base Station Antenna Arrays: Spectrum Efficiency, Algorithms and Propagation Models*. PhD thesis, Royal Institute of Technology, Stockholm, Sweden, June 1997.

[23] Per Zetterberg and Björn Ottersten. The spectrum efficiency of a base station antenna array for spatially selective transmission. *IEEE Transactions on Vehicular Technology*, 44:651–660, August 1995.

An Overview of A Simulation Environment to Study the Impact of Non-Ideal Hardware on Array Processing

J. Yin, C.M.S. See, B.P. Ng, Y.K. Sim and Y.L. Lu *

Invited Paper

Abstract

This paper presents an antenna array system simulation environment. The simulation environment comprises of a multipath signal model, an antenna array model and a multichannel receiver model. Collectively, they provide a simulation framework to analyze the impact of non-ideal hardware on high performance array processing. The simulation environment is developed with Signal Processing Workstation (SPW) and Matlab.

1 Introduction

Antenna array processing plays an integral role in communications, radar and surveillance operations, such as signal detection, direction finding and beamforming (waveform recovery). Most of the array processing algorithms developed for signal detection, direction finding and beamforming rely mostly on the following mathematical model:

$$\mathbf{x}(t) = \sum_{i=1}^{d} \mathbf{a}(\theta_i) s_i(t) + \mathbf{n}(t) \tag{1}$$

where $\mathbf{x}(t)$ is the complex envelope representation of the antenna array output, $\mathbf{a}(\theta_i)$ is the steering vector (or antenna response vector) and $\mathbf{n}(t)$ denotes the noise corrupting the received signals. With on this model, numerous

---

*J. Yin, B.P. Ng and Y.L. Lu are with School of EEE, Nanyang Technological University. C.M.S. See and Y.K. Sim are with DSO National Laboratories, 20 Science Park Drive, SINGAPORE 118230

algorithms have been developed for signal detection and parameter estimation over the past few decades, such as [3] [4] [5]. When the antenna array system output matches the mathematical model in (1), the full potential of these algorithms, such as the super-resolution direction finding, can be realized. However, this is not true in practice. Possible hardware errors that may depart from the received signal from the ideal model in (1) include

- gain and phase stability, and non-linearities of amplifiers,
- finite precision and non-linearities in analog-to-digital converter (ADC),
- phase and amplitude jitters and drift due to limited stability of oscillators,
- uncertainties in array geometry due to finite mechanical precision,
- uncertainties in elemental antenna pattern due to antenna array calibration errors and nearby scatterers (especially non-stationary scatterers).

Many performance analysis on the effects of these non-idealities have been conducted. However, most of them analyze these effects either in isolation or based on gross error [6] [9] [8] which offer limited applications in fine-grain array processing system analysis, design and specifications. In fine grain system analysis, the errors introduced by the antenna array and the multichannel receiver are analyzed the sub-systems (and modules) level. Such analysis will enable us to achieve the following:

- Study the impact of the non-ideal effects introduced by a sub-system on the array processing algorithm, in presence of errors introduced other sub-systems.
- Establish the interrelationships of the (non-ideal) characteristics and their effect on the array processing algorithms.
- Analyze and identify critical hardware parameters on array processing algorithms.
- Enable a closely coupled array processing algorithm and hardware co-design.
- Establish the system requirement, and perform cost-performance optimization via exploration of appropriate system parameters and architectures.

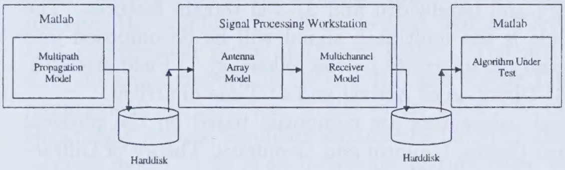

Matlab
Multipath Propagation Model
Signal Processing Workstation
Antenna Array Model
Multichannel Receiver Model
Matlab
Algorithm Under Test
Harddisk
Harddisk

Figure 1: Antenna Array Processing Simultion Environment.

In addition to robust specification on the antenna array system's sub-modules, such analysis can also enable the design of systems parsimonious from cost and volume viewpoints. Such fine grain analysis is complicated and is likely to be achievable using modeling and simulation.

In view of its potential usefulness, since 1998, DSO National Laboratories and School of EEE, Nanyang Technological University have collaborated under the MINDEF-NTU Applied R & D MOU to jointly develop a computer simulation environment based on SPW and Matlab that will enable fine-grain antenna array system analysis, design and specifications. This paper presents the current state of development and its future directions.

2 Simulation Environment

The simulation environment comprises of the following models:Multipath Propagation Model, Antenna Array Signal Model and Multichannel Receiver Model. The functional relationships of these module are depicted Figure 1. These models will generate the time-domain vectored signal as inputs to the array processing algorithm under test. Next, we shall examine each model in detail.

2.1 Multipath Propagation Model

This module generates the signal propagation parameters based on the transmitter-receiver geometry, radio parameters and terrain features. It is implemented in Matlab. The inputs of this module are: 1) transmitter and receiver location, 2) signal frequency and bandwidth and 3) 3-D terrain features. The output from this module is the multipath signal will be decomposed into individual signal path and parametrized by the following: 1) Field strength, 2) Polarization state, 3) Direction of Arrival and 4) Time of arrival.

The multipath signal parameters are computed based on the physical principles of Geometrical Optics, Uniform and Geometric Theory of Diffraction. Given the 3-D terrain database, these physical theories are amalgamated to provide the framework to determine all the possible raypaths and their field properties. We will not present the theoretical development of this multipath propagation module here. For further details, we refer readers to [2].

2.2 Antenna Array Signal Model

This module generates the observed signal at the output of the antenna array in the following manner. Based on the direction of arrival parameters from the multipath propagation model and given the source signal waveform, the time-delayed version of the signal due to its propagation across the antenna arrival is given by

$$s\left(t-\tau_{i}\right)=\operatorname{IFT}\left(\alpha\left(f\right) S\left(f\right) \exp\left(j 2 \pi f \tau_{i}\right)\right) \tag{2}$$

where τ_i is the differential time-delay of the signal $s\left(t\right)$ at the i^{th} antenna with respect to a reference point. $\operatorname{IFT}(\cdot)$ is the inverse Fourier transform operator, $\alpha\left(f\right)$ is the path loss at frequency f and $S\left(f\right)$ is the Fourier transform of $s\left(t\right)$. Given the impulse response of the i^{th} antenna, $h_i\left(t,\gamma\right)$, the time-domain observed signal at the i^{th} antenna is given by

$$y_{i}\left(t\right)=h_{i}\left(t, \gamma\right) \star s\left(t-\tau_{i}\right) \tag{3}$$

Note that the equations (2) and (3) do not limit their application to narrowband signal. The impulse response of the i^{th} antenna, $h_i\left(t,\gamma\right)$can be derived from antenna prediction packages such as NEC. Currently, we are investigating on incorporating the effects of dynamically varying local scatters.

2.3 Multichannel Receiver Model

The multichannel receiver is designed based on commercially available off the shelves components. The multichannel receiver is modelled based on the following basic functional modules: 1) amplifiers, 2) mixers, 3) filters, and 4) analog to digital converter.

2.3.1 Amplifier and Filter

A key requirement in our amplifier model is the ability to define and parameterize the amplifier based on its gain, 2nd and 3rd order intercept point, 1dB compression point and noise figure. The non-linear amplifier block <rf_lib/amp.symbol> of SPW's RF and Radar Library meets this basic requirement. In addition, it also offers table look-up feature which allows empirically derived amplifier response be incorporated into the multichannel receiver model. This is an important feature when we need to examine the effects of the amplifier at various gain settings. Such analysis can provide the necessary insights on the requirement on receiver calibration over the various gain settings in the receiver chain. We use <rf_lib/amp.symbol> as are basic building amplifier block in our multichannel receiver model.

Filtering is an important operations in receivers. Their performance will determine the ability to reject image and out of band signals, as well as the level of signal degradation due to aliasing. The amplitude and phase response of the filters and their degree of mismatch across the receiver channels will determine the (frequency) resolution required to achieve the desired performance in array processing. Hence, the choice of filter can impact on the complexity and performance of the array processing system. In our multichannel receiver model, we tapped on the wide variety of filter blocks in SPW's DSP Library for the bandpass, highpass, lowpass and bandstop filtering operations.

2.3.2 Mixer

Mixing operations in receiver convert the signal at radio frequency to an appropriate intermediate frequency before they are processed by the analog to digital converter. In practical receivers, the signal at the mixer output is typically degraded by frequency, amplitude and phase jitters of the local oscillator, while non-linearities are introduced due to its product operation. These effects can have significant impact on the performance of most array processing algorithms.

Figure 2 depicts the block diagram of the mixer model used in our multichannel receiver model. We use SPW's RF and Radar Library mixer model <rf_lib/mix_noise.symbol> and extend it to include local oscillator's frequency and phase jitters. Currently, we are expanding this model to include effects due to amplitude jitter.

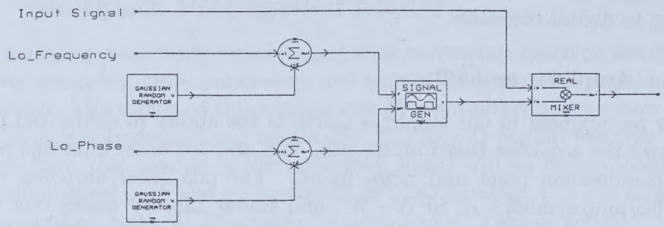

Input Signal
Lo_Frequency
Lo_Phase
GAUSSIAN RANDOM GENERATOR
GAUSSIAN RANDOM GENERATOR
SIGNAL
GEN
REAL
MIXER

Figure 2: Mixer model incorporating the effects of phase and frequency jitter.

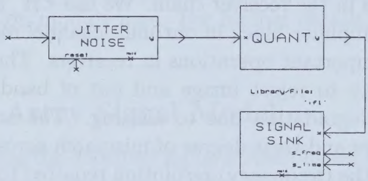

JITTER NOISE
reset
QUANT
Library/File:
SIGNAL SINK
s_freq
s_time

Figure 3: ADC model.

2.3.3 Analog to Digital Converter

The analog to digital converter (ADC) quantizes the signal in time and amplitude of analog signals into digital samples. Amplitude quantization will introduce quantization noise due to the finite amplitude discretization of the signal. The signal to noise ratio of the signal after the ADC process will be degraded further by jitters in the timing clocks. In addition, the quality of the sampled signal can be further degraded by non-linear effects due to ADC saturation, as well as, the dynamic properties of the device itself.

Figure 3 depicts the ADC model currently in used in our simulation model. It comprises of the quantizer and jitter noise block. The former provides a model for quantization error and ADC saturation, while the latter introduces the effects of timing jitter during the ADC process. Presently, we are working towards incorporating non-linear response (input-output) to the current model.

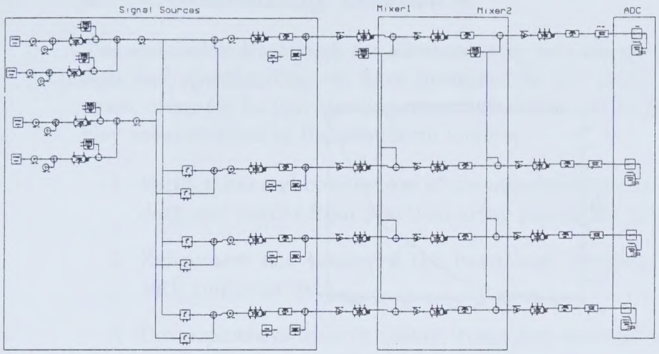


Figure 4: Antenna Array Processing System Simulation Environment.

3 A Simulation Example

Next, we shall present an example of an antenna array processing system analysis based on the proposed simulation environment. The antenna array processing system operates at 2GHz with baseband bandwidth of 500kHz, It uses 4 antenna element and adopts an uniform linear array geometry. It uses a double down conversion receiver architecture with local oscillators at 1930MHz and 69MHz. The corresponding intermediate frequency are 70MHz and 1MHz. The simulation block diagram of the antenna array processing system is shown in Figure 4. It is developed with the basic multichannel receiver building blocks that were introduced in Section 2.3.

The plot in Figure 5 shows the impact of timing jitter during analog to digital conversion on direction of arrival estimation. Figure 6 illustrate the effect of signal level at the input of ADC on the direction of arrival estimation. The results indicate that the antenna array processing system can achieve relative good DF performance with 10 and 12 bit ADC as long as gain < 1. When saturation occurs (i.e. gain >1), their DF performance degrade rapidly. The 8 bit ADC does not seems to be an appropriate choice for the antenna array processing system. At gain $<< 1$, poor DF performance is observed. This is probably due to the manifestation of quantization noise.

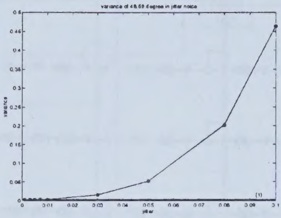


Figure 5: Effects of ADC Timing Jitter on DF with Capon's beamformer. Jitter is defined as a ratio to sampling period.

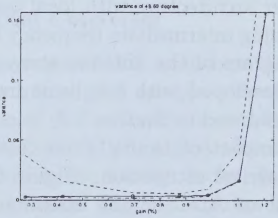


Figure 6: DF performance versus Gain. Gain is defined as the ratio of signal level to its full scale. 12, bit ADC '__', 10 ADC bit: '_._' and 8 ADC bit: '- - -'.

4 Concluding Remark

The simulation framework for antenna array processing system analysis, design and specifications we have presented is still at its infancy. There is much room for further development and refinement of the various models. Key areas of focus in the near term include:

1. Verification and comparison of the simulation results against empirical data and results from practical array processing systems.
2. Refinement and tuning of the functional modules (and sub-modules) with empirical data.
3. Development of tools to enable "computer-assisted" antenna array processing system design and optimization.

References

[1] S. Haykin, "Advances in Spectrum Analysis and Array Processing, Volume III", Prentice Hall, New Jersey.

[2] K.Y. Kam and L.Y. Chew, "A Deterministic Multipath Propagation Signal Model", MINDEF-NUS Joint R&D Seminar, 1998.

[3] R.O. Schmidt,"Multiple Emitter Location and Signal Parameter Estimation", *Proc RADC Spectrum Estimation Workshop*, Rome, NY, 1979, pp. 243-258.

[4] M. Wax, "Detection and Estimation of Superimposed Signals", *Ph.D. Thesis, Stanford University,* 1985.

[5] R.H. Roy, "ESPRIT, Estimation of Signal Parameters via Rotational Invariance Technique", *Ph.D. Thesis, Stanford University,* 1987.

[6] B. Friedlander, "A Sensitivity Analysis of the MUSIC Algorithm", *IEEE Trans. Acoustics, Speech and Signal Processing.*

[7] F. Li and R. Vaccaro, "Sensitivity Analysis of DOA Estimation Algorithms to Sensor Errors", *IEEE Trans. on Aerospace and Electronic Systems,* Vol. 28, pp. 708-717, 1992.

[8] M. Viberg and A.L. Swindlehurst, "Analysis of the Combined Effects of Finite Samples and Model Errors on Array Processing Performance", *IEEE Trans. on Signal Processing,* Vol. 42, pp. 3073-3083, 1994.

Signal Representations for Transmit-Receive Antenna Arrays*

Jianzhong Zhang, Kraiyut Tantinarawat and Akbar M. Sayeed
Department of Electrical and Computer Engineering
University of Wisconsin–Madison
Madison, WI 53706
zjianzho@cae.wisc.edu, tantinar@cae.wisc.edu, akbar@engr.wisc.edu

Abstract

We propose a signal representation for transmit-receive uniform linear arrays operating over arbitrary time-varying spatial multipath channels encountered in mobile communications. The signal representation decomposes the multipath channel into a finite number of *virtual channels* that correspond to certain delays, Doppler shifts, angles that are *fixed a priori* and are independent of the physical channel parameters. Temporal channel variations are captured *linearly* by the variations in the fading coefficients associated with the virtual channels. The utility of the canonical signal representation is illustrated with simple transceiver structures. The signal representation entails simple channel estimation procedures and can be leveraged for a variety of physical layer functions, including diversity signaling, spatially-selective multiuser signaling, and space-time coding.

1 Introduction

The use of antenna arrays at both the transmitter and receiver can significantly improve communication over the mobile wireless channel. In particular, there has been significant interest in the development of space-time coding techniques for multiple-element transmit arrays to enhance the bandwidth efficiency of the wireless link. However, most existing techniques rely on simplistic channel models — most assume uncorrelated or weakly dependent correlation between different elements and ignore channel dispersion effects [1, 2, 3]. Knowledge of channel characteristics can result in improved performance and a more realistic space-time multipath model is developed in [4]. However the channel modeling in [4] requires the estimation of several time-varying channel parameters, such as delays and directions of arrival (DOAs), which often entail nonlinear parameter estimation algorithms.

In this paper, we propose a simple yet fairly general approach for modeling time-varying space-time multipath channels. Our formulation builds on the canonical channel modeling introduced in [5] for the case of a single transmitter and multiple receivers to develop a complete model for transmit-receive antenna arrays. An attractive feature of our formulation is that it decomposes the overall multipath channel into a finite number of *virtual channels* corresponding to certain multipath delays, Doppler shifts and angles that are *fixed a priori* and are independent of the actual delays and DOAs encountered during propagation. Furthermore, all temporal channel variations are captured linearly by the variations in the fading coefficients associated with the virtual channels. The canonical space-time signal model provides a useful framework for transceiver design for spatial multipath channels. In particular, it can be leveraged for

---

[*] This research is supported in part by Wisconsin Alumni Research Foundation and by NSF under Grant No. CCR-9875805.

enabling space-time coding techniques that rely on uncorrelated coupling between different transmit-receiver sensors.

In the next section, we present a signal representation for the transmitter array. In Section 3, we develop the canonical signal model for transmit-receive antenna arrays operating over time-varying multipath channels. Some useful alternative representations of the model are presented in Section 4. Section 5 discusses simple transceiver structures based on the canonical model to illustrate its utility. Concluding remarks are provided in Section 6.

2 Transmit Beamforming

A transmit array of N_T sensors is capable of sending N_T different signals with independent spatial profiles. We focus on a one-dimensional N_T-element uniform linear array (ULA) with spacing d. Let $q_1(t), q_2(t), \ldots, q_{N_T}(t)$ be the N_T information bearing signals of duration T with corresponding spatial profiles given by $\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_{N_T}$. The N_T-dimensional signal transmitted by the sensor array, $\mathbf{x}(t) = [x_1(t), x_2(t), \ldots, x_{N_T}(t)]^T$, is given by

$$\mathbf{x}(t) = \sum_{n=1}^{N_T} \mathbf{a}_n q_n(t) = \mathbf{A}\mathbf{q}(t),\ 0 \leq t \leq T, \tag{1}$$

where $\mathbf{q}(t) = [q_1(t), q_2(t), \ldots, q_{N_T}(t)]^T$ is the vector of information bearing signals and $\mathbf{A}$ is the matrix of spatial profile vectors $\mathbf{A} = [\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_{N_T}]$. As illusrated in Figure 1, the far-field spatial distribution $y(t, \phi)$ due to the transmitted signal is the superposition of the signals transmitted from all the sensors and is given by

$$y(t, \phi) = \sum_{n=1}^{N_T} e^{\frac{-j2\pi(n-1)d\sin\phi}{\lambda}} x_n(t) = \mathbf{a}^H(\phi)\mathbf{x}(t) \tag{2}$$

where $\mathbf{a}(\phi)$ denotes the *array steering vector* corresponding to the spatial direction ϕ given by

$$\mathbf{a}(\phi) = [1, e^{\frac{j2\pi d\sin(\phi)}{\lambda}}, \ldots, e^{\frac{j2\pi(N_T-1)d\sin(\phi)}{\lambda}}]^T. \tag{3}$$

If we do a change of variable $\phi \rightarrow \xi : \xi = \frac{\sin\phi}{\lambda}$ the above equation becomes

$$\begin{aligned}\tilde{y}(t, \xi) &= y(t, \sin^{-1}(\xi\lambda)) = \sum_{n=1}^{N_T} e^{-j2\pi(n-1)d\xi} x_n(t) = \tilde{\mathbf{a}}^H(\xi)\mathbf{x}(t) \\ &= \tilde{\mathbf{a}}^H(\xi)\mathbf{A}\mathbf{q}(t) = \mathbf{f}^H(\xi)\mathbf{q}(t) = \sum_{n=1}^{N_T} f_n(\xi) q_n(t)\end{aligned} \tag{4}$$

where

$$\tilde{\mathbf{a}}(\xi) = \mathbf{a}(\sin^{-1}(\xi\lambda)),\ \ \mathbf{f}^H(\xi) = [f_1(\xi), f_2(\xi), \ldots, f_{N_T}(\xi)],\ \ \text{and} f_n(\xi) = \tilde{\mathbf{a}}^H(\xi)\mathbf{a}_n \tag{5}$$

Note that $f_n(\xi)$ denotes the far-field spatial response in the direction ξ due to the transmit spatial profile $\mathbf{a}_n$.

3 Canonical Signal Modeling for Multipath Channels

Time-varying multipath propagation is characteristic of mobile wireless channels. Figure 1 presents a simple one-dimensional picture of the coupling between the transmit array and the receive array through spatial multipath scattering. $\tilde{y}(t, \xi)$ denotes the spatial distribution at the scatterer due to the transmitted signal. $\tilde{z}(t, \zeta)$ is the spatial distribution seen by the receive

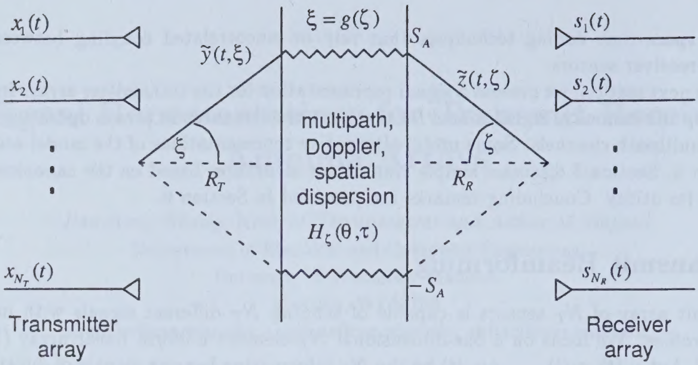


Figure 1: Transmitter Antenna Array

array. Notice that there is a one-to-one mapping between the transmit spatial coordinate ξ and receive spatial coordinate ζ. The mapping depends on the relative geometry of the transmit array, receive array, and the scattering medium. In the far-field configuration, the following approximate relation holds

$$\xi = g(\zeta) \approx \frac{R_R}{R_T}\zeta \tag{6}$$

where R_R and R_T denote the distance from the scatterer to the receive and transmit array, respectively.

In addition to this remapping of the spatial coordinate, the spatial signal seen by the receive array is also distorted by the scattering medium. Such effects include spatial distortion, temporal dispersion due to multipath effects, and spectral (Doppler) dispersion due to temporal variations. We model these scattering effects via a *spatial multipath-Doppler spreading function*, $H_\xi(\theta,\tau)$, represented in the transmit spatial coordinate ξ in Figure 1 [5]. The spatio-temporal signal $\tilde{z}(t,\zeta)$ seen by the receive array is given by [5]

$$\tilde{z}(t,\zeta) = \int_0^{S_M}\int_{-S_D}^{S_D} H_{g(\zeta)}(\theta,\tau)\tilde{y}(t-\tau, g(\zeta))e^{j2\pi\theta t}d\theta d\tau \tag{7}$$

where S_M and S_D denote the multipath and Doppler spreads of the wireless channel, respectively.[1] The information bearing signal vector $\mathbf{s}(t)$ at the N_R-dimensional receive array is given by [5]

$$\begin{aligned}
\mathbf{s}(t) &= \int_{-S_A}^{S_A}\tilde{z}(t,\zeta)\tilde{\mathbf{a}}(\zeta)d\zeta = \int_{-S_A}^{S_A}\int_0^{S_M}\int_{-S_D}^{S_D} H_{g(\zeta)}(\theta,\tau)\tilde{y}(t-\tau, g(\zeta))\tilde{\mathbf{a}}(\zeta)e^{j2\pi\theta t}d\theta d\tau d\zeta \\
&= \int_{-S_A}^{S_A}\int_0^{S_M}\int_{-S_D}^{S_D} H_{g(\zeta)}(\theta,\tau)\tilde{\mathbf{a}}(\zeta)\mathbf{f}^H(g(\zeta))\mathbf{q}(t-\tau)e^{j2\pi\theta t}d\theta d\tau d\zeta \\
&= \sum_{n=1}^{N_T}\int_{-S_A}^{S_A}\int_0^{S_M}\int_{-S_D}^{S_D} H_{g(\zeta)}(\theta,\tau)f_n(g(\zeta))\tilde{\mathbf{a}}(\zeta)q_n(t-\tau)e^{j2\pi\theta t}d\theta d\tau d\zeta
\end{aligned} \tag{8}$$

where S_A denotes the *angular* channel spread seen by the receiver. The above expression shows the coupling between each transmitted waveform $q_n(t)$ and the received signal $\mathbf{s}(t)$ through the corresponding spreading function

$$H_{n,g(\zeta)}(\theta,\tau) = H_{g(\zeta)}(\theta,\tau)f_n(g(\zeta)). \tag{9}$$

---

[1] We note that the S_M and S_D could vary with angle. We assume that they represent the maximum values and the variations with ξ are accounted for in $H_\xi(\theta,\tau)$.

It is shown in [5] that due to the finite duration T and essentially finite bandwidth of the waveforms $\{q_n(t)\}$, the received signal $\mathbf{s}(t)$ exhibits only a finite number of temporal degrees of freedom corresponding to uniformly spaced discrete multipath delays and Doppler shifts associated with each $q_n(t)$. Adapting the derivation for a single transmit sensor in [5] to the model (8), we get the following canonical representation for the N_R-dimensional signal at the receiver array

$$\mathbf{s}(t) \approx \mathbf{s}_C(t) = \sum_{n=1}^{N_T} \sum_{l=0}^{L} \sum_{m=-M}^{M} \sum_{p=-P}^{P} h_{npml} \mathbf{u}_{npml}(t) \tag{10}$$

where the basis waveforms $\mathbf{u}_{npml}(t)$ are given by

$$\mathbf{u}_{npml}(t) = \tilde{\mathbf{a}}(\varphi_p) q_n \left(t - \frac{l}{B}\right) e^{j\frac{2\pi m t}{T}} \tag{11}$$

and the expansion coefficients h_{npml} are samples, in (ζ, θ, τ), of a smoothed version of $H_{n,g(\zeta)}(\theta, \tau)$. The discrete angles, Doppler shifts and delays, $(\zeta, \theta, \tau) = (\varphi_p, m/T, l/B)$, define the canonical basis waveforms and are independent of the actual physical channel parameters. The canonical directions φ_p satisfy $\varphi_p = \frac{p}{N_T d_T}$ where d_T is the sensor spacing at the transmitter [5, 7].

The representation (10) decomposes the channel into *virtual channels* defined by the basis waveforms (11). The basis waveforms are fixed *a priori* and their number depends on the multipath, Doppler and angular channel spreads. All temporal variations in physical channel characteristics are captured by the canonical channel coefficients $\{h_{npml}\}$. We note that timing/angular offsets can be easily incorporated into the representation (10) to address channel acquisition issues.

4 Alternative Input-Output Representations

The representation (10) can also be written in the multiple-input multiple-output form

$$\mathbf{s}_C(t) = \begin{bmatrix} s_{C,1}(t) \\ \vdots \\ s_{C,N_R}(t) \end{bmatrix} = \mathbf{A}_C \mathbf{H}_C \mathbf{V}_C(t) \mathbf{q}_C(t) \tag{12}$$

which provides an intuitive interpretation in terms of the canonical delays, Doppler shifts and directions. The vector

$$\mathbf{q}_C(t) = \left[q_1(t), \ldots, q_1\left(t - \frac{L}{B}\right), \ldots, q_{N_T}(t), \ldots, q_{N_T}\left(t - \frac{L}{B}\right)\right]^T$$

consists of all the canonical multipath delays (within the multipath spread) of each transmitted waveform, $q_n(t)$, that can possibly appear at the receiver. The matrix

$$\mathbf{V}_C(t) = \mathbf{I}_{(L+1)N} \otimes \left[e^{-j\frac{2\pi M t}{T}}, \ldots, e^{j\frac{2\pi M t}{T}}\right]^T,$$

where $\otimes$ denotes Kronecker product, generates all possible canonical Doppler shifts of the waveforms in $\mathbf{q}_C(t)$. The matrix

$$\mathbf{H}_C = \begin{bmatrix} h_{-P,1,0,-M} & \cdots & h_{-P,1,0,M} & \vdots & h_{-P,1,L,-M} & \cdots & h_{-P,1,L,M} & \vdots & h_{-P,N_T,L,M} \\ \vdots & & \vdots & \vdots & \vdots & & \vdots & \vdots & \vdots \\ h_{P,1,0,-M} & \cdots & h_{P,1,0,M} & \vdots & h_{P,1,L,-M} & \cdots & h_{P,1,L,M} & \vdots & H_{P,N_T,L,M} \end{bmatrix} \tag{13}$$

consists of the canonical channel coefficients and combines the canonical multipath-Doppler basis waveforms, for each $q_n(t)$, to generate the signal seen by the receiver in each canonical spatial direction. Each row of $\mathbf{H}_C$ corresponds to a particular canonical received angle and consists of the multipath-Doppler channel coefficients for each transmitted waveform associated with that angle. Finally, the canonical received signal representation, $\mathbf{s}_C(t)$, is a linear combination of the array responses (generated by $\mathbf{A}_C$) to the signals from different canonical directions. The canonical receiver *array response matrix* $\mathbf{A}_C$ is given by

$$\mathbf{A}_C = [\tilde{\mathbf{a}}(\varphi_{-P}), \ldots, \tilde{\mathbf{a}}(\varphi_P)] \tag{14}$$

In order to analyze the performance of transceivers based on the representation (10), it is advantageous to represent in an equivalent form in which the channel coefficient matrix $\mathbf{H}_C$ is vectorized. Following [7], we can express $s_C(t)$ as

$$\mathbf{s}_C(t) = \mathbf{U}_C \mathbf{Q}_C(t) \mathbf{h}_C, \text{ where} \tag{15}$$

$$\begin{aligned}
\mathbf{Q}_C(t) &= diag\{\psi_C(t)\} \otimes \mathbf{I}_{2P+1}, \\
\psi_C(t) &= \left[e^{-j\frac{2\pi M t}{T}}, \ldots, e^{j\frac{2\pi M t}{T}}\right]^T \otimes \left[q_1(t), \ldots, q_1\left(t - \frac{L}{B}\right), \ldots, q_{N_T}(t), \ldots, q_{N_T}\left(t - \frac{L}{B}\right)\right]^T, \\
\mathbf{U}_C &= ones(1, (L+1)(2M+1)(N_T)) \otimes \mathbf{A}_C, \text{ and} \\
\mathbf{h}_C &= [h_{1,-M,0,-P}, \ldots, h_{N_T,M,L,P}]^T .
\end{aligned}$$

5 Transceiver Structure

In this section, we describe some simple transceiver structures based on the canonical signal model. The canonical channel decomposition (10) into *virtual channels*, defined by the basis waveforms (11), can be exploited in a variety of ways. For example, the virtual channels can be used as diversity channels to combat fading, as multiple channels to increase the data rate, or as increased dimensions for coding. We start by investigating the number on independent degrees of freedom in the channel revealed by the canonical signal representation.

5.1 Degrees of Freedom

The number of degrees of freedom in the space-time channel depends on the nature of the propagation medium. To study this relationship, we use a discrete multipath model for the propagation characteristics that is often used in practice: the transmitted signal propagates over different paths, each corresponding to a particular DOA (seen at the receiver), delay and fading coefficient.[2] In our notation, the actual received signal can be modeled as

$$\mathbf{s}(t) = \sum_{l=1}^{L_o} \beta_l \tilde{\mathbf{a}}(\zeta_l) \left[\sum_{n=1}^{N_T} f_n(g(\zeta_l)) q_n(t - \tau_l)\right] \tag{16}$$

where L_o is the total number of (dominant) paths, $f_n(\xi)$ is the far-field spatial signature for the n^{th} waveform defined in (5), and $\tau_l \in (0, S_M)$, $\zeta_l \in (-S_A, S_A)$, β_l denote the delay, DOA and fading coefficient, respectively, for the l^{th} path. All these channel parameters vary with time in a mobile environment. The above *physical* signal model can be compactly written as

$$\mathbf{s}(t) = \mathbf{A}_P \mathbf{Q}_P(t) \tilde{\beta} \tag{17}$$

where $\mathbf{A}_P = ones(1 \times N_T) \otimes [\tilde{\mathbf{a}}(\zeta_1), \ldots, \tilde{\mathbf{a}}(\zeta_{L_o})]$

$$\mathbf{Q}_P(t) = diag\{q_1(t-\tau_1), \ldots, q_1(t-\tau_{L_o}), \ldots, \ldots, q_{N_T}(t-\tau_1), \ldots, q_{N_T}(t-\tau_{L_o})\}, \text{ and} \tag{18}$$

---

[2] The Doppler effects arise primarily from the temporal variations in the fading coefficients. We ignore Doppler effects for simplicity.

$$\tilde{\beta} = [f_1(g(\zeta_1))\beta_1, \ldots, f_1(g(\zeta_{L_o}))\beta_{L_o}, \ldots, \ldots, \ldots, f_{N_T}(g(\zeta_1))\beta_1, \ldots, f_{N_T}(g(\zeta_{L_o}))\beta_{L_o}]^T . \quad (19)$$

The received signal $\mathbf{r}(t)$ at the array is given by $\mathbf{r}(t) = \mathbf{s}(t) + \mathbf{n}(t)$ where $\mathbf{n}(t)$ denotes additive white Gaussian noise.

The dependence of the degrees of freedom on the propagation characteristics can be illustrated by focusing on the spatial aspects only (ignoring multipath delays). In this case, the noise-free canonical signal representation is $\mathbf{s}_C(t) = \sum_{n=1}^{N_T} \sum_{p=-P}^{P} h_{np}\mathbf{u}_{np}(t)$ where the canonical basis functions are given by $\mathbf{u}_{np}(t) = \tilde{\mathbf{a}}(\varphi_p)q_n(t)$. In a receiver based on this canonical representation, the front-end processing corresponds to projecting the received signal onto the basis waveforms $\{\mathbf{u}_{np}(t)\}$. The signal component of these matched filter outputs is given by[3]

$$z_{n,p} = \int_0^T \mathbf{u}_{n,p}^H(t)\mathbf{s}(t)dt = \sum_{l=1}^{L_o} \beta(\zeta_l) f_n(g(\zeta_l))\tilde{\mathbf{a}}^H(\varphi_p)\tilde{\mathbf{a}}(\zeta_l) = \sum_{l=1}^{L_o} \beta(\zeta_l) f_n(g(\zeta_l))c_p(\zeta_l) = \mathbf{v}_{n,p}^T\beta \quad (20)$$

where $c_p(\zeta_l) = \tilde{\mathbf{a}}^H(\varphi_p)\tilde{\mathbf{a}}(\zeta_l)$, $\mathbf{v}_{n,p} = [f_n(g(\zeta_1))c_p(\zeta_1), \ldots, f_n(g(\zeta_{L_o}))c_p(\zeta_{L_o})]^T$, and $\beta = [\beta(\zeta_1), \ldots, \beta(\zeta_{L_o})]^T$. Putting $\{z_{n,p}\}$ into a vector we get $\mathbf{z} = \mathbf{F}\beta$ where the $N_T(2P+1) \times L_o$ matrix $\mathbf{F}$ consists of $\{\mathbf{v}_{n,p}^T\}$ as its rows. The number of independent degrees of freedom in the space-time channel, as manifested in the canonical model, are governed by the rank of the correlation matrix of $\mathbf{z}$ $\mathbf{R}_z = \mathrm{E}[\mathbf{z}\mathbf{z}^H] = \mathbf{F}\mathbf{R}_\beta\mathbf{F}^H$. Clearly rank$(\mathbf{R}_z) \leq \min(N_T(2P+1), L_o)$. With weak dependence between different propagation paths, $\mathbf{R}_\beta$ can be full (L_o) rank. Furthermore, for $L_o \geq N_T(2P+1)$, $N_T(2P+1)$ independent degrees of freedom can be achieved. Since $2P+1 \leq N_R$, the maximum number of independent degrees of freedom is $N_T N_R$. We note that with the same sensor spacing at both the transmitter and receiver, at most $N_T + N_R - 1$ independent channels can be created. However, the maximum number can be achieved with mismatched spacings.

5.2 Single Data Stream

In this section, we analyze a transceiver structure that uses all degrees of freedom for diversity to combat fading; that is, the same bit stream is transmitted on all independent channels. We analyze performance for coherent antipodal signaling. The received signal vector $\mathbf{r}(t)$ is given by $\mathbf{r}(t) = b\mathbf{s}(t) + \mathbf{n}(t)$ where $\mathbf{s}(t)$ is the information bearing signal, $b \in \{-1, 1\}$ is the information bit, and $\mathbf{n}(t)$ is the N_R dimensional complex Gaussian vector with zero mean and $\mathrm{E}[\mathbf{n}(t)\mathbf{n}^H(t')] = \mathcal{N}_0\delta(t-t')\mathbf{I}_{N_R}$.

The optimal test statistic is obtained by projecting the received signal onto the canonical basis functions, $z_{npml} = \langle \mathbf{r}, \mathbf{u}_{npml}\rangle$, and doing maximal-ratio-combining (MRC)

$$Z = Re\left\{ \sum_{n=1}^{N_T} \sum_{l=0}^{L} \sum_{m=-M}^{M} \sum_{p=-P}^{P} h_{npml}^* z_{npml} \right\} = Re\{\mathbf{h}_C^H\mathbf{z}\}. \quad (21)$$

The vector $\mathbf{z}$ of matched filter outputs $\{z_{npml}\}$ is given by

$$\mathbf{z} = \int_0^T \mathbf{Q}_C^H(t)\mathbf{U}_C^H\mathbf{r}(t)dt = b\left(\int_0^T \mathbf{Q}_C^H(t)\mathbf{U}_C^H\mathbf{A}_P\mathbf{Q}_P(t)\tilde{\beta}dt\right) + \int_0^T \mathbf{Q}_C^H(t)\mathbf{U}_C^H\mathbf{n}(t)dt \quad (22)$$

where we have used the physical (disrete multipath) signal model (16). We compute the average bit-error probability (P_e) assuming perfect estimates of canonical channel coefficients $\{h_{npml}\}$. Following [7], define cross-correlation matrix between the canonical and physical waveforms as

$$\mathbf{R}_{CP} = \int_0^T \mathbf{Q}_C^H(t)\mathbf{U}_C^H\mathbf{A}_P\mathbf{Q}_P(t)dt \quad (23)$$

---

[3] We assume orthogonal $\{q_n(t)\}$ for simplicity. Only linear independence is needed in general.

and the auto-correlation matrix between the canonical basis waveforms as

$$\mathbf{R}_{CC} = \int_0^T \mathbf{Q}_C^H(t)\mathbf{U}_C^H\mathbf{U}_C\mathbf{Q}_C(t)dt. \tag{24}$$

The test statistic (21) can then be written as

$$Z = Re\left\{b\mathbf{h}_C^H\mathbf{R}_{CP}\tilde{\beta} + \mathbf{h}_C^H\mathbf{w}\right\} \tag{25}$$

where $\mathbf{w} = \int_0^T \mathbf{Q}_C^H(t)\mathbf{U}_C^H\mathbf{n}(t)dt$ and $\mathrm{E}[\mathbf{w}\mathbf{w}^H] = \mathcal{N}_0\mathbf{R}_{CC}$. The canonical channel coefficients estimated from a noise-free pilot are given by $\mathbf{h}_C = \mathbf{R}_{CC}^{-1}\mathbf{R}_{CP}\tilde{\beta}$ [7]. Hence, (25) becomes

$$Z = b\tilde{\beta}^H\mathbf{R}_{CP}^H\mathbf{R}_{CC}^{-1}\mathbf{R}_{CP}\tilde{\beta} + Re\{\tilde{\beta}^H\mathbf{R}_{CP}^H\mathbf{R}_{CC}^{-1}\mathbf{w}\}. \tag{26}$$

Conditioned on the fading vector $\tilde{\beta}$ defined in (19), $P_e(\tilde{\beta})$ is given by

$$P_e(\tilde{\beta}) = Q\left(\sqrt{\frac{2}{\mathcal{N}_0}\tilde{\beta}^H\mathbf{R}_{CP}^H\mathbf{R}_{CC}^{-1}\mathbf{R}_{CP}\tilde{\beta}}\right) \tag{27}$$

The average bit-error probability can be obtained by averaging the $P_e(\tilde{\beta})$ over a χ^2 distribution induced by the Gaussian fading coefficients $\{\beta_l\}$ in (16). Assuming independent, equal-power fading coefficients $\{\beta_l\}$, we have $E[\beta\beta^H] = \mathcal{E}I/L_o$, where $\mathcal{E}$ is the total received energy. Using (19), the covariance matrix of $\tilde{\beta}$ is then given by $\mathbf{R}_{\tilde{\beta}} = E[\tilde{\beta}\tilde{\beta}^H] = \mathcal{E}I/L_o\mathbf{G}\mathbf{G}^H$, where

$$\mathbf{G} = \begin{bmatrix} f_1(g(\zeta_1)) & \cdots & 0 \\ 0 & \ddots & 0 \\ 0 & \cdots & f_1(g(\zeta_{Lo})) \\ \vdots & \vdots & \vdots \\ f_{N_T}(g(\zeta_1)) & \cdots & 0 \\ 0 & \ddots & 0 \\ 0 & \cdots & f_{N_T}(g(\zeta_{Lo})) \end{bmatrix} \tag{28}$$

defines the relationship between the physical fading coefficent vector β and the sptially-warped fading coefficient vector $\tilde{\beta}$ (see (19)). It follows that P_e is given by

$$P_e = \frac{1}{2}\sum_{l=1}^{L_e}\prod_{i=1,i\neq l}^{L_e}\frac{\lambda_l}{\lambda_l - \lambda_i}\left(1 - \sqrt{\frac{\rho\lambda_l}{\rho\lambda_l + 1}}\right) \tag{29}$$

where $\rho = \mathcal{E}/\mathcal{N}_0L_o$, $\{\lambda_l\}$ are the non-zero eigenvalues of the matrix $\mathbf{\Phi} = \mathbf{R}_{CP}^H\mathbf{R}_{CC}^{-1}\mathbf{R}_{CP}\mathbf{G}\mathbf{G}^H$ and L_e is the number of the non-zero eigenvalues for matrix $\mathbf{\Phi}$. The number of effective diversity channels are determined by the number of significant $\{\lambda_l\}$ and their relative magnitudes.

5.3 Multiple Data Streams

In this section we analyze a transceiver structure in which the degrees of freedom in the channel are split between multiple data streams. Let there be K bit streams. The k^{th} bit stream is associated with N_k transmit signaling waveforms $\{q_n(t)\}$ such that $\sum_k N_k = N_T$. In this case, the physical received signal can be modeled as

$$\begin{aligned} \mathbf{r}(t) &= \sum_{l=1}^{L_o}\beta_l\tilde{\mathbf{a}}(\zeta_l)\left[\sum_{n=1}^{N_T} f_n(g(\zeta_l))q_n(t-\tau_l)b_n\right] \\ &= \mathbf{s}(t) + \mathbf{n}(t) = \mathbf{A}_P\mathbf{Q}_P(t)\mathbf{H}_P\mathbf{b} + \mathbf{n}(t) \end{aligned} \tag{30}$$

where

$$\mathbf{b} = [\ b_1 \quad b_2 \quad \ldots \quad b_K\ ]^T, \text{ and}$$

$$\mathbf{H}_P = diag\{\mathbf{h}_1, \mathbf{h}_2, \ldots, \mathbf{h}_K\}, \text{ where}$$

$$\mathbf{h}_k = \mathbf{G}_k\beta = \left[\beta_1 f_{N'_k+1}(g((\zeta_1)), \ldots, \beta_{L_o} f_{N'_k+1}(g(\zeta_{L_o})), \ldots, \beta_1 f_{N'_k+N_k}(g(\zeta_1)), \ldots, \beta_{L_o} f_{N'_k+N_k}(g((\zeta_{L_o}))\right]^T \tag{31}$$

where $N'_1 = 0$, $N'_k = \sum_{i=1}^{k-1} N_i$, $k = 2, \ldots, K$, and $\mathbf{G}_k$ is the corresponding k^{th} sub-block of $\mathbf{G}$ defined in (28). The corresponding canonical representation of the noise-free signal is

$$\mathbf{s}_C(t) = \mathbf{U}_C \mathbf{Q}_C(t) \mathbf{H}_C \mathbf{b} \tag{32}$$

where

$$\mathbf{H}_C = diag\{\mathbf{h}_{C1}, \mathbf{h}_{C2}, \ldots, \mathbf{h}_{CK}\}, \quad \text{and}$$

$$\mathbf{h}_{Ck} = [h_{N'_k+1,-M,0,-P}, \ldots, h_{N'_k,M,L,P}, \ldots, h_{N'_k+N_k,-M,0,-P}, \ldots, h_{N'_k+N_k,M,L,P}]^T$$

In this case, for simplicity, we also represent the actual received signal in the canonical representation to compute the test statistics. The physical model is incorporated in computing the canonical channel coefficients for MRC. Thus, the matched filter test statistics are given by

$$\mathbf{z} = \langle \mathbf{r}, \mathbf{u}_{npml} \rangle = \mathbf{R}_{CC} \mathbf{H}_C \mathbf{b} + \mathbf{n} \tag{33}$$

where $\mathbf{n} \sim N(\mathbf{0}, \mathcal{N}_0 \mathbf{R}_{CC})$. In this case, the different bit streams interfere with each other due to $\mathbf{R}_{CC}$ in (33), and maximum-likelihood (ML) bit estimates are needed for optimal performance. However, ML estimation can be computationally expensive and we resort to linear suboptimal approaches analogous to those employed in multiuser detection. We first obtain an estimate of the interference-free correlator outputs, $\mathbf{y} = \mathbf{H}_C \mathbf{b}$, and then do MRC to obtain the bit estimate for each data stream. A linear estimator is employed: $\hat{\mathbf{y}} = \mathbf{L}\mathbf{z}$. There are several choices of the estimation matrix $\mathbf{L}$. Here we adopt the decorrelating solution $\mathbf{L} = \mathbf{R}_{CC}^{-1}$. It follows that the estimate of $\mathbf{y}$ is $\hat{\mathbf{y}} = \mathbf{R}_{CC}^{-1}\mathbf{z} = \mathbf{H}_C\mathbf{b} + \tilde{\mathbf{n}}$ where $\tilde{\mathbf{n}} \sim N(0, \mathcal{N}_0 \mathbf{R}_{CC}^{-1})$. Now, MRC can be applied to the correlator outputs corresponding to each data stream $\mathbf{y}_k$ to obtain the bit estimates. However, since the noise in the estimate $\hat{\mathbf{y}}_k$ is correlated, a prewhitening operation on each $\hat{\mathbf{y}}_k$ is needed prior to MRC. The overall multi-data stream receiver takes the form

$$\hat{\mathbf{b}} = sign\{Re[\mathbf{H}_C^H \mathbf{D} \hat{\mathbf{y}}]\} \tag{34}$$

where the block diagonal matrix $\mathbf{D} = diag\{\mathbf{Q}_{11}^{-1}, \mathbf{Q}_{22}^{-1}, \ldots, \mathbf{Q}_{KK}^{-1}\}$ is the prewhitening matrix defined in terms of the sub-blocks of the noise correlation matrix $\mathbf{Q} = \mathbf{R}_{CC}^{-1}$, and the matrix $\mathbf{H}_C^H$ performs MRC.

From the overall receiver structure (34), the test statistic corresponding to the k^{th} data stream is

$$w_k = \mathbf{h}_{Ck}^H \mathbf{Q}_{kk}^{-1} \mathbf{h}_{Ck} b_k + \mathbf{h}_{Ck}^H \mathbf{Q}_{kk}^{-1} \tilde{\mathbf{n}}_k \tag{35}$$

Conditioned on the knowledge of the canonical channel coefficients $\mathbf{h}_{Ck}$, the bit-error probability for the k^{th} stream is

$$P_k(\mathbf{h}_{Ck}) = Q\left(\sqrt{\frac{2}{\mathcal{N}_0} \mathbf{h}_{Ck}^H \mathbf{Q}_{kk}^{-1} \mathbf{h}_{Ck}}\right) = Q\left(\sqrt{\frac{2}{\mathcal{N}_0} \|\tilde{\mathbf{h}}_{Ck}\|^2}\right) \tag{36}$$

where $\tilde{\mathbf{h}}_{Ck} = \mathbf{Q}_{kk}^{-1/2} \mathbf{h}_{Ck}$. The unconditional bit-error probability can be obtained by averaging $P_k(\mathbf{h}_{Ck})$ over the density function of $\tilde{\mathbf{h}}_{Ck}$:

$$P_k = \frac{1}{2} \sum_{l=1}^{L_e} \prod_{i=1, i \neq l}^{L_e} \frac{\lambda_l}{\lambda_l - \lambda_i} \left(1 - \sqrt{\frac{\lambda_l}{1 + \lambda_l}}\right) \tag{37}$$

where $\{\lambda_l\}$ are the eigenvalues of the correlation matrix $\mathbf{R}_{\tilde{\mathbf{h}}} = 1/\mathcal{N}_0 E[\tilde{\mathbf{h}}_{Ck}\tilde{\mathbf{h}}_{Ck}^H] = 1/\mathcal{N}_0(\mathbf{Q}_{kk}^{-1} E[\mathbf{h}_{Ck}\mathbf{h}_{Ck}^H]$ and L_e is the number of non-zero eigenvalues. In order to compute the correlation matrix of $\mathbf{h}_{Ck}$, we need to find the relationship between $\mathbf{h}_{Ck}$ and $\mathbf{h}_k$, or more generally $\mathbf{H}_C$ and $\mathbf{H}_P$. In the case of perfect channel estimation from a noise-free pilot, the relation is

$$\mathbf{H}_C = \arg \min_{\mathbf{H}_C} ||\mathbf{U}_C \mathbf{Q}_C(t)\mathbf{H}_C \mathbf{b} - \mathbf{A}_P \mathbf{Q}_P(t)\mathbf{H}_P \mathbf{b}||^2 = \mathbf{R}_{CC}^{-1}\mathbf{R}_{CP}\mathbf{H}_P, \tag{38}$$

and thus, $\mathbf{h}_{Ck}$ is related to $\mathbf{h}_k$ via

$$\mathbf{h}_{Ck} = [\mathbf{R}_{CC}^{-1}\mathbf{R}_{CP}]_{kk}\mathbf{h}_k = \Psi_{kk}\mathbf{h}_k \tag{39}$$

where $\Psi_{kk} = [\mathbf{R}_{CC}^{-1}\mathbf{R}_{CP}]_{kk}$ is the k^{th} diagonal block of $\mathbf{R}_{CC}^{-1}\mathbf{R}_{CP}$, with $\mathbf{h}_k$ given by $\mathbf{h}_k = \mathbf{G}_k\beta$ in (31). Following the development in the single bit-stream case, we assume all the L_o discrete paths are independent and have the same power. With this assumption, the aforementioned eigenvalues are given by $\lambda_l = \frac{\mathcal{E}}{\mathcal{N}_0 L_o} eig_l \left[\mathbf{Q}_{kk}^{-1}\Psi_{kk}\mathbf{G}_k\mathbf{G}_k^H\Psi_{kk}^H\right]$.

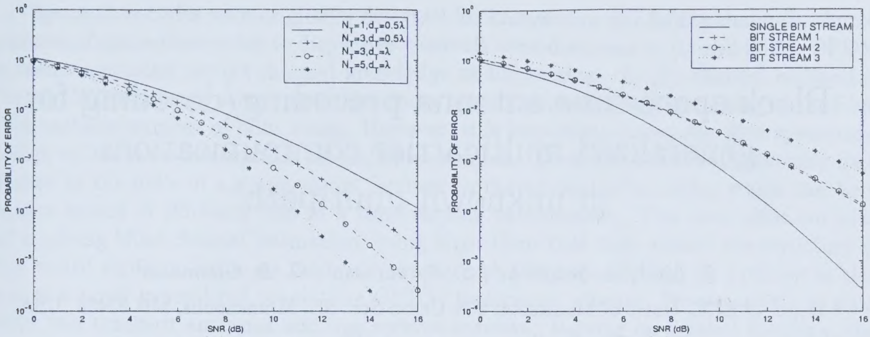
5.4 Numerical Examples

In this section, we present some numerical examples illustrating the transceiver structures analyzed above. A transmit-receive uniform receive array with $N_R = 3$ receiver elements and different ($N_T = 1, 3, 5$) transmit elements is considered. A spatial channel with angular spread $S_A = -S_A = \pi/5$ is assumed with zero delay spread ($S_M = 0$). The physical signal (16) corresponds to $L_o = 41$ paths with DOAs uniformly distributed on $[-\pi/5, \pi/5]$. The canonical receiver is based on 3 beams with uniformly spaced angles. Figure 2(a) depicts the performance of canonical transceiver structure for the single data-stream case. The receiver sensor spacing is $d_R = \lambda/2$. Transceiver performance is compared for different number of transmitters ($N_T = 1, 3, 5$) and different transmitter spacings $d_T = d_R = 0.5\lambda$ and $d_T = 2d_R = \lambda$. The total transmitted power is held constant in all cases. It is evident that the performance improves with the number of transmitters due to additional diversity. Furthermore, keeping N_T constant, the performance also improves with mismatched spacing due to the higher level of achievable independent degrees of freedom (see Section 5.1).

Figure 2(b) depicts the performance for an $N_T = N_R = 3$ element transmit-receive array when three different bit-streams are transmitted on each transmit signaling waveform $q_n(t)$. The performance is compared with the case when a single stream is sent on all three transmit channels. The transmitted power in all three streams in the multiple stream case is the same as the transmitted power in the single stream case (thus, total transmitted power is 3 times higher). We note that the performance of the different streams in the 3-stream case is roughly equivalent to the performance of the $N_T = 1$ case in Figure 2(a) since each stream gets about 3 diversity channels. However, the performance is poorer than the single-stream performance for $N_T = 3$ due to the lower level of diversity. We note that this performance comparison is based on simple *uncoded* transmission on the three streams. Space-time coding at the transmitter can yield significantly better performance than either of the two extreme cases.

6 Conclusions

In this paper, we have proposed a canonical signal model for transmit-receive uniform linear arrays operating over time-varying multipath channels. Our model decomposes the overall spatio-temporal multipath channel into virtual channels defined by fixed space-time basis waveforms that are independent of the physical channel delays and DOAs. All temporal channel variations are captured by the variations in the fading coefficients associated with the virtual channels. In addition to leading to simple channel estimation algorithms, the canonical signal model can be leveraged for enabling space-time coding techniques that assume uncorrelated coupling between different transmit-receive elements. The virtual basis waveforms could also lead to efficient coded modulation techniques appropriate for spatial multipath fading channels.



(a) (b)

Figure 2: Performance of simple transceiver structures based on the canonical signal model. (a) Single data-stream. Performance comparison for different N_T and different transmitter sensor spacing. (b) Multiple data-streams. Each stream has the same transmitted power as the single stream in (a). The performance of each stream is about the same as for $N_T = 1$ case in (a) but worse than the $N_T = 3$ case due to lower level of diversity.

References

[1] G. J. Foschini, "Layered space-time architecture for wireless communication in a fading environment when using multi-element antennas," *Bell Labs Tech. J.*, vol. 1, no. 2, pp. 41–59, 1996.

[2] G. J. Foschini and M. J. Gans, "On limits of wireless communications in a fading environment when using multiple antennas," *Wireless Personal Communications*, vol. 6, pp. 311–335, 1998.

[3] V. Tarokh, N. Seshadri, and A. R. Calderbank, "Space-time codes for high data rate wireless communication: Performance criteria and code construction," *IEEE Trans. Inform. Th.*, pp. 744–765, Mar. 1998.

[4] G. G. Rayleigh and T. Boros, "Joint space-time parameter estimation for wireless communication channels," *IEEE Trans. Signal Processing*, pp. 1333–1343, May 1998.

[5] A. M. Sayeed, E. N. Onggosanusi, and B. D. Van Veen, "A canonical space-time characterization of mobile wireless channels," *IEEE Communications Letters*, pp. 94–96, Apr. 1999.

[6] E. N. Onggosanusi, A. M. Sayeed, and B. D. Van Veen, "Canonical space-time processing in CDMA systems," in *Proc. 1999 IEEE Int. Conf. on Acoust., Speech and Signal Proc. — ICASSP '99.*

[7] E. N. Onggosanusi, A. M. Sayeed, and B. D. Van Veen, "Canonical space-time processing in wireless communications," *submitted to the IEEE Trans. Commun.*, Apr. 1999.

Block space-time antenna precoding/decoding for generalized multicarrier communications in unknown multipath*

Z. Liu [1], *A. Scaglione*[1], *S. Barbarossa*[2], *G. B. Giannakis*[1]

[1] Dept. of ECE, Univ. of Minnesota, 200 Union Str. SE, Minneapolis, MN 55455, USA,
E-mails: [lzq, anna, georgios]@ece.umn.edu

[2] INFOCOM Dept., Univ. of Rome "La Sapienza", Via Eudossiana 18, 00184 Rome, Italy,
E-mail: sergio@infocom.ing.uniroma1.it

Abstract

Transmit antenna diversity has been exploited recently to develop high performance space-time coders and simple maximum-likelihood decoders for transmissions over known flat fading channels. Relying on block precoding, this paper develops generalized space-time coded multicarrier transceivers appropriate for wireless propagation over frequency-selective (and perhaps unknown) multipath channels. Multicarrier precoding maps the frequency-selective channel into a set of flat fading sub-channels, whereas space-time encoding/decoding facilitates equalization and achieves performance gains by exploiting the diversity available with transmit antennas. When unknown, channel status information is acquired blindly based on a deterministic variant of the constant-modulus algorithm that capitalizes on block precoding. System performance is evaluated both analytically and with simulations.

1 Introduction

In recent years, space-time coding has gained much attention as an effective transmit diversity technique to combat fading in wireless communications. In transmit diversity schemes relying on space-time codes, channel coding that exploits multiple transmit antennas is combined with appropriate signal processing at the receiver to provide diversity gain. Space-time trellis codes were first proposed in [9] to achieve maximum diversity gain. However, for a fixed number of transmit antennas, decoding complexity at the receiver increases exponentially with the transmission rate. To reduce the decoding complexity, space-time block codes with two transmit antennas were first introduced by Alamouti [2] and later generalized to an arbitrary number of transmit antennas in [10]. An attractive property of space-time block codes is that maximum-likelihood decoding at the receiver can be performed using only linear processing. Interestingly, for complex constellations, space-time block coding with two transmit antennas is the only block code to provide full diversity without loss of transmission rate [10].

---

*This research is supported in part by ARO grant no. DAAG55-98-1-0336

Space-time codes were originally designed for known slow flat fading channels. Applications of space-time codes to dispersive channels were discussed in [1] and [4] for OFDM systems assuming perfect channel knowledge at the receiver. In [3], channel estimation for an OFDM scheme using space-time codes was implemented using training in the form of a suitable number of pilot tones. However, it is important to remark that space-time codes utilize state information on multiple channels. Thus, the achievable diversity gain comes at the price of a proportional increase in the amount of training, which can be a major source of efficiency loss in a time-varying environment. This motivates our idea of applying blind channel estimation, using algorithms that fully exploit the structure of the coded signal to estimate channel parameters. More specifically, we propose in this paper a novel generalized space-time OFDM transceiver scheme. We consider a system with two transmit antennas and one receive antenna. Relying on symbol blocking, the space-time block encoder using two transmit antennas [2] is incorporated into the generalized OFDM transceiver scheme [8] to achieve transmit diversity in frequency-selective propagation. In addition to performance improvement, the decoding simplicity of the space-time block encoder of [2] is retained in our system, even when communicating over frequency-selective channels. By exploiting the special structure of space-time block codes, direct blind channel estimation based on a deterministic variant of the constant-modulus algorithm is also investigated in this paper.

The paper is organized as follows. In Section 2, we describe the system model and the transceiver design. In Section 3, we present our blind channel estimation algorithm. Based on our design, bit-error-rate performance is analyzed in Section 4 and extensive simulations are presented in Section 5. Section 6 concludes this paper.

Notation: Column vectors are denoted by boldface lower case letters; matrices are denoted by boldface capital letters. The superscripts ^{T}, *, $^{\mathcal{H}}$ stand for transpose, complex conjugate and complex conjugate transpose, respectively.

2 System Modeling and Design

Consider a wireless system with two transmit antennas at the base station and only one receive antenna. We will concentrate on transmissions from the base station to the mobile users, i.e., on the forward link. Fig. 1 depicts the chip-rate sampled discrete-time equivalent baseband system model. Similar to a conventional OFDM transmission scheme, the information sequence $s(n)$ is parsed in blocks $\mathsf{s}(n) := [s(nK), \ldots, s(nK + K - 1)]^T$ of size $K \times 1$. Our space-time encoder maps every two consecutive symbol blocks $\mathsf{s}(2n)$ and $\mathsf{s}(2n+1)$ to the following $2K \times 2$ matrix:

$$\begin{bmatrix} \mathsf{s}(2n) & -\mathsf{s}^*(2n+1) \\ \mathsf{s}(2n+1) & \mathsf{s}^*(2n) \end{bmatrix}, \tag{1}$$

whose columns are transmitted in successive time intervals with the upper and lower blocks sent through the first and second antenna element respectively. Note that without blocking, (1) shows the same structure as space-time block codes with two transmit antennas [2]. Prior to transmission the symbol block in each antenna branch is mapped onto a P-long block with $P > K$ through a linear transformation described by the tall $P \times K$ matrix $\mathbf{C}$. After parallel-to-serial (P/S) conversion and modulation (not shown in Fig. 1), the transmitted symbols propagate through FIR channels, $\{h_i(l),\ l = 0, \ldots, L\}$, $i = 1, 2$, where L denotes an upper bound on the maximum channel order which is assumed to be available (the channel response includes transmit-receive pulse shaping filters, multipath, and relative delay between the two antennas). Setting the last $P - K := L_g$ rows of $\mathbf{C}$ to zero and taking $L_g \geq L$, interblock interference (IBI) is avoided, so that we can focus on each received block separately. The channels are completely described by the $P \times P$ Toeplitz matrices $\mathbf{H}_i$, $i = 1, 2$, shown in Fig. 1. After S/P conversion (not shown in Fig. 1), each pair of two consecutive blocks $\mathbf{y}(n)$ and $\mathbf{y}(n+1)$, with $\mathbf{y}(n) := [y(nP), \ldots, y(nP + P - 1)]^T$, received from the two transmit antennas is given by:

$$\begin{aligned} \mathbf{y}(2n) &= \mathbf{H}_1\mathbf{C}\mathbf{s}(2n) + \mathbf{H}_2\mathbf{C}\mathbf{s}(2n+1) + \mathbf{w}(2n) \\ \mathbf{y}(2n+1) &= -\mathbf{H}_1\mathbf{C}\mathbf{s}^*(2n+1) + \mathbf{H}_2\mathbf{C}\mathbf{s}^*(2n) + \mathbf{w}(2n+1), \end{aligned} \tag{2}$$

where $\mathbf{w}(n) := [w(nP), \ldots, w(nP + P - 1)]^T$ denotes AWGN.

Our basic idea is to achieve space-time coding and decoding in the frequency ($\mathcal{Z}$-) domain by judicious design of $\mathbf{C}$. Define the $\mathcal{Z}$-transforms: $Y(n; z) = \sum_{p=0}^{P-1} y(nP+p)z^{-p}$; $C_k(z) := \sum_{p=0}^{P-1} c_k(p + 1)z^{-p}$, where $c_k(p)$ is (p, k)th entry of $\mathbf{C}$; $H_i(z) = \sum_{l=0}^{L} h_i(l)z^{-l}$; $W(n; z) = \sum_{p=0}^{P-1} w(nP + p)z^{-p}$. Let us focus on the $\mathbf{y}(2n)$ first. It follows from (2) that

$$\begin{aligned} Y(2n; z) &= \sum_{\mu=1}^{K} [H_1(z)C_\mu(z)s(2nK + \mu - 1) + H_2(z)C_\mu(z)s((2n+1)K + \mu - 1)] \\ &+ W(2n; z). \end{aligned} \tag{3}$$

We seek code polynomials $C_\mu(z)$, $\mu = 1, \ldots, K$, which guarantee that there exist at least K distinct points $\{\rho_k\}_{k=1}^{K}$ such that for each k, $Y(2n; \rho_k)$ contains the contribution from $s(2nK+k-1)$ and $s((2n+1)K+k-1)$ only, irrespective of $H_1(z)$ and $H_2(z)$. To achieve this goal, we design $C_\mu(\rho_k)$ to satisfy

$$C_\mu(\rho_k) = A\delta(k - \mu), \ \forall \mu, k \in [1, K], \tag{4}$$

where A is a constant chosen to impose the transmission power constraint. If such codes exist to satisfy condition (4), plugging ρ_k into (3) we obtain $\forall k \in [1, K]$

$$Y(2n; \rho_k) = AH_1(\rho_k)s(2nK + k - 1) + AH_2(\rho_k)s((2n + 1)K + k - 1) + W(2n; \rho_k). \tag{5}$$

The same code polynomial design allows us to isolate from $Y(2n + 1; z)$ through $\{\rho_k\}_{k=1}^{K}$ the signals $s^*((2n + 1)K + k - 1)$ and $s^*(2nK + k - 1)$, i.e., $\forall k$

$$\begin{aligned} Y(2n + 1; \rho_k) &= -AH_1(\rho_k)s^*((2n + 1)K + k - 1) + AH_2(\rho_k)s^*(2nK + k - 1) \\ &+ W(2n + 1; \rho_k). \end{aligned} \tag{6}$$

For simplicity of notation, let us define the $P \times 1$ Vandermonde vector $\mathbf{v}(\rho, P)$ built from the complex constant ρ as: $\mathbf{v}(\rho, P) := [1, \ \rho^{-1}, \ldots, \rho^{-(P-1)}]^T$. With z replacing ρ, the $\mathcal{Z}$-transform of any $P \times 1$ vector $\mathbf{x}(n)$ can be represented by $X(n; z) = \mathbf{v}^T(z, P)\mathbf{x}(n)$. Choosing the filterbank described by the $P \times K$ matrix

$$\mathbf{V} := [\mathbf{v}(\rho_1, P), \ldots, \mathbf{v}(\rho_K, P)]^T, \tag{7}$$

we observe that $Y(n; \rho_k)$, $k = 1, \ldots, K$, are the outputs of the filterbank $\mathbf{V}$.

Define $\tilde{\mathbf{y}}_k(n) := [Y(2n; \rho_k), \ Y^*(2n + 1; \rho_k)]$. It follows from (5) and (6) that

$$\tilde{\mathbf{y}}_k(n) = A \ \tilde{\mathbf{H}}_k \ \mathbf{s}_k(n) + \tilde{\mathbf{w}}_k(n), \ \forall k \in [1, K], \tag{8}$$

where $\mathbf{s}_k(n) := [s(2nK+k-1),\ s((2n+1)K+k-1)]^T$, $\tilde{\mathbf{w}}_k(n) := [W(2n;\rho_k),\ W^*(2n+1;\rho_k)]^T$ and

$$\tilde{\mathbf{H}}_k := \begin{bmatrix} H_1(\rho_k) & H_2(\rho_k) \\ H_2^*(\rho_k) & -H_1^*(\rho_k) \end{bmatrix}. \tag{9}$$

Note that (8) reveals an important feature of our design: $2K$-long block transmissions over frequency selective fading channels have been converted to K parallel 2-long block transmissions over flat fading channels, so that the complexity of channel estimation and signal detection can be reduced considerably. Observe also that $\tilde{\mathbf{H}}_k$ in (9) is a unitary matrix as in [2]. Hence, the Zero-Forcing (ZF) equalizer for $\mathbf{s}_k(n)$ can be described by the matrix $\mathbf{D}_k := \tilde{\mathbf{H}}_k^{\mathcal{H}}$, $\forall k \in [1,\ K]$. After decoding $\mathbf{s}_k(n)$, information sequence $s(n)$ can be retrieved by the P/S operation shown in Fig. 1.

Let us now design code polynomials $\{C_\mu(z)\}_{\mu=1}^K$ to satisfy (4). For fixed μ, (4) prescribes $C_\mu(z)$ at K points ρ_k. Thus, polynomials $C_\mu(z)$ that satisfy (4) should have degree $\deg[C_\mu(z)] \geq K-1$. When $\deg[C_\mu(z)] = K-1$, polynomials $C_\mu(z)$ can be uniquely determined by Lagrange interpolation through the points $\{\rho_k\}_{k=1}^K$ as in [8]:

$$C_\mu(z) = A\,\frac{\prod_{k=1,k\neq\mu}^K (1-\rho_k z^{-1})}{\prod_{k=1,k\neq\mu}^K (1-\rho_k \rho_\mu^{-1})}, \quad \mu \in [1,K]. \tag{10}$$

Using (10) and taking into account the $L_g = L$ guard zeros, the code length in our system is $P = K + L$. Since we deal with transmissions of K-long blocks, the bandwidth efficiency of our system is

$$\eta := \frac{K}{K+L}. \tag{11}$$

Note that for sufficiently large K, we have $\eta \simeq 1$; hence, bandwidth is not over expanded in our system.

Let us now consider a specific choice of the points $\{\rho_k\}_{k=1}^K$. Specifically, let ρ_k's be chosen regularly spaced around the unit circle on the complex plane, i.e.,

$$\rho_k = e^{j\frac{2\pi}{K}(k-1)}, \ \forall k \in [1,K]. \tag{12}$$

It can be shown that the corresponding matrix $\mathbf{C}$ coincides with the IDFT matrix, with $A = \sqrt{K}$. Accordingly, matrix $\mathbf{V}$ in (7) becomes the DFT matrix. Thus, matrix multiplication by $\mathbf{C}$ and $\mathbf{V}$ can be replaced by FFTs which have very low computational complexity. Hence, we deduce that OFDM is a special case of our design.

3 Blind Channel Estimation

We further assume transmission of constant modulus (CM) symbols (e.g., we use M-PSK modulation). Using our transceiver scheme of Section 2, we show next how to estimate the two channel impulse responses $h_1(l)$ and $h_2(l)$ *blindly* by exploiting the CM property of the transmitted symbols. Consider the data model in (8) and simplify the notation by absorbing A into $\tilde{\mathbf{H}}_k$ and ρ_k in the subscript k, to rewrite it as:

$$\tilde{\mathbf{y}}_k(n) = \tilde{\mathbf{H}}_k \mathbf{s}_k(n) + \tilde{\mathbf{w}}_k(n), \tag{13}$$

where $\tilde{\mathbf{y}}_k(n) := [Y_{1k}(n),\ Y_{2k}^*(n)]^T$, $\mathbf{s}_k(n) := [s_{1k}(n),\ s_{2k}(n)]^T$ and

$$\tilde{\mathbf{H}}_k := \begin{bmatrix} H_{1k} & H_{2k} \\ H_{2k}^* & -H_{1k}^* \end{bmatrix}. \tag{14}$$

Starting from (13), we first take without loss of generality [1], $|s_{1k}(n)|^2 = 1$, $|s_{2k}(n)|^2 = 1$, because all blind methods yield channel estimates up to a constant factor. Hence, the problem we consider in this section is: given received data $\tilde{\mathbf{y}}_k(n)$ only, find equalizer $\mathbf{G}_k$ such that $\hat{\mathbf{s}}_k(n) := \mathbf{G}_k\tilde{\mathbf{y}}_k(n)$ has entries with unit constant modulus. Note that the input $\mathbf{s}_k(n)$ need not be white. This problem is equivalent to a generalized eigenvalue problem as presented in [11]. However, we develop here a simpler algorithm which takes advantage of the specific structure of $\tilde{\mathbf{H}}_k$.

Let us consider blind channel estimation in high SNR cases where noise $\tilde{\mathbf{w}}_k(n)$ is negligible (noise effects will be tested in the simulations). Due to the unitary structure of $\tilde{\mathbf{H}}_k$ and because $\mathbf{G}_k\tilde{\mathbf{H}}_k = \mathbf{I}$, we obtain the following structure for $\mathbf{G}_k$:

$$\mathbf{G}_k := \begin{bmatrix} g_{1k} & g_{2k} \\ g_{2k}^* & -g_{1k}^* \end{bmatrix} = \frac{1}{|H_{1k}|^2 + |H_{2k}|^2} \begin{bmatrix} H_{1k}^* & H_{2k} \\ H_{2k}^* & -H_{1k} \end{bmatrix}. \tag{15}$$

Writing $\hat{\mathbf{s}}_k(n) := [\hat{s}_{1k}(n),\ \hat{s}_{2k}(n)]^T = \mathbf{G}_k\tilde{\mathbf{y}}_k(n)$ component-wise and imposing the CM constraint $|\hat{s}_{1k}(n)|^2 = 1$, $|\hat{s}_{2k}(n)|^2 = 1$, we arrive at $\mathbf{Y}_k(n)\mathbf{g}_k = [1,\ 1]^T$, where

$$\mathbf{Y}_k(n) := \begin{bmatrix} |Y_{1k}(n)|^2 & |Y_{2k}(n)|^2 & Y_{1k}(n)Y_{2k}(n) & Y_{1k}^*(n)Y_{2k}^*(n) \\ |Y_{2k}(n)|^2 & |Y_{1k}(n)|^2 & -Y_{1k}(n)Y_{2k}(n) & -Y_{1k}^*(n)Y_{2k}^*(n) \end{bmatrix}, \ \mathbf{g}_k := \begin{bmatrix} g_{1k}g_{1k}^* \\ g_{2k}g_{2k}^* \\ g_{1k}g_{2k}^* \\ g_{1k}^*g_{2k} \end{bmatrix}.$$

Next, we stack N blocks of data $\{\tilde{\mathbf{y}}(n)\}_{n=0}^{N-1}$ and concatenate $\mathbf{Y}_k(n)\mathbf{g}_k$ to find

$$\mathbf{Y}_k\mathbf{g}_k = [1, \ldots, 1]_{2N\times 1}^T \tag{16}$$

where $\mathbf{Y}_k := [\mathbf{Y}_k^T(0), \ldots, \mathbf{Y}_k^T(N-1)]^T$. If $\mathbf{Y}_k$ is full column rank, the pseudo-inverse of $\mathbf{Y}_k$ will give a unique solution of $\mathbf{g}_k$ from (16). Unfortunately, it can be proved that the maximum column rank of $\mathbf{Y}_k$ is only three (see also [11]). Nevertheless, the solution of (16) can still be sought in the form $\mathbf{g}_k = \mathbf{g}_{0k} + \beta\mathbf{g}_{1k}$, where $\mathbf{g}_{0k}$ is a particular solution of (16) and $\mathbf{g}_{1k}$ spans the kernel of $\mathbf{Y}_k$ which has dimension 1. The coefficient β can be determined by imposing $\mathbf{g}_k(1)\mathbf{g}_k(2) = \mathbf{g}_k(3)\mathbf{g}_k(4)$ where $\mathbf{g}_k(i)$ denotes the ith entry of $\mathbf{g}_k$. Solving (16) for $\mathbf{g}_k$ and then using (15) we can recover the magnitude of the two channels and the sum of their phase as: $\forall k \in [1, K]$

$$\begin{aligned} |H_{ik}| &= |H_i(\rho_k)| = \frac{\sqrt{\mathbf{g}_k(i)}}{\mathbf{g}_k(1) + \mathbf{g}_k(2)}, \qquad i = 1, 2 \\ \angle\, H_{1k} + \angle\, H_{2k} &= \angle\, H_1(\rho_k) + \angle\, H_2(\rho_k) = \angle\, \mathbf{g}_k(4). \end{aligned} \tag{17}$$

However, since the solution of $\mathbf{g}_k(1)\mathbf{g}_k(2) = \mathbf{g}_k(3)\mathbf{g}_k(4)$ leads to a second-order equation in β, we end up with two possible solutions, which leads to an ambiguity between the two rows of matrix $\mathbf{G}_k$ and, as a consequence, between the two channels.

As an example, in Fig. 2 we show the two channels' magnitude estimates obtained blindly at SNR=15 dB, using the previous method. We can see that indeed this simple method yields accurate estimates of the channel magnitude, but we have ambiguity between the two channels. To remove this ambiguity a precoder described by the matrix $\mathbf{F}$ is introduced at the transmitter, and consequently the data model in (13) is rewritten as

$$\tilde{\mathbf{y}}_k(n) = \tilde{\mathbf{H}}_k\mathbf{F}\mathbf{s}_k(n) + \tilde{\mathbf{w}}_k(n), \tag{18}$$

---

[1] We deal with complex modulations only in this paper. The extension to real modulations, e.g., BPSK, is straightforward and the constraints to be imposed are $s_{1k}^2(n) = 1$ and $s_{2k}^2(n) = 1$.

where $\mathbf{F}$ is chosen to be a 2×2 full-rank diagonal real matrix denoted as $\mathbf{F} := \mathrm{diag}(d_1,\ d_2)$, $d_1, d_2 \neq 0$. Accordingly, we reformulate the problem of searching for the $\mathbf{G}_k$ with the constraint that each entry of $\hat{\mathbf{s}}_k(i) := \mathbf{F}^{-1}\mathbf{G}_k\tilde{\mathbf{y}}_k(i)$ has unit constant modulus. By imposing such a CM constraint over N blocks, we obtain

$$\mathbf{T}\mathbf{Y}_k\mathbf{g}_k = [1, \ldots, 1]^T_{2N\times1} \tag{19}$$

where $\mathbf{T} := \mathrm{diag}(1/d_1^2, 1/d_2^2, \ldots, 1/d_1^2, 1/d_2^2)$ and $\mathbf{Y}_k$ is built as in (16). It can be proved that when $d_1 \neq d_2$, $\mathbf{Y}_k$ in (19) is full rank with probability 1 for N sufficiently large [5], and thus $\mathbf{g}_k$ can be uniquely determined by inverting the linear system in (19). As already mentioned, the knowledge of $\mathbf{g}_k$ allows us to estimate the two channels' magnitude and the sum of their phase. At this point, the only parameter left to determine is the phase of one of the two channel transfer functions. This reduces the number of real channel parameters to be estimated and correspondingly the number of necessary training bits.

Looking for a fully blind solution we propose next a subspace algorithm that exploits a typical assumption made on wireless communication channels, upon which OFDM systems also rely: the finite support of the channel impulse response. For each K, from (17) we obtain $r(k) := H_1^*(\rho_k)/H_2(\rho_k)$, $k = 1, \ldots, K$. Enforcing the finite channel support, we also find

$$\mathbf{v}^T(\rho_k^*, L)\ \mathbf{h}_1^* - r(k)\ \mathbf{v}^T(\rho_k, L)\ \mathbf{h}_2 = 0, \quad \forall k \in [1, K], \tag{20}$$

where $\mathbf{h}_i := (h_i(0), \ldots, h_i(L))^T$, for $i = 1, 2$. The homogeneous equations in (20) can be cast in the following matrix form

$$\Theta \begin{bmatrix} \mathbf{h}_1^* \\ \mathbf{h}_2 \end{bmatrix} = 0, \tag{21}$$

where

$$\Theta := \begin{bmatrix} \mathbf{v}^T(\rho_1^*, L) & -r(1)\mathbf{v}^T(\rho_1, L) \\ \vdots & \vdots \\ \mathbf{v}^T(\rho_K^*, L) & -r(K)\mathbf{v}^T(\rho_K, L) \end{bmatrix}.$$

If $\mathbf{h}_1$, $\mathbf{h}_2$ do not have common zeros and the channel orders are known, it can be proved that (21) provides a unique solution for $\mathbf{h}_1$ and $\mathbf{h}_2$ within a scalar ambiguity, which can be eliminated using only one pilot tone.

4 Performance Analysis

Taking into account the precoder $\mathbf{F}$, we rewrite the data model in (8) as

$$\tilde{\mathbf{y}}_k(n) = A\tilde{\mathbf{H}}_k\mathbf{F}\mathbf{s}_k(n) + \tilde{\mathbf{w}}_k(n),\ \forall k \in [1, K]. \tag{22}$$

Because $w(n)$ is AGN, $W(n; \rho_k)$ is also AGN. Theoretical BER evaluation is thus possible for a given constellation if we adopt a ZF equalizer. From (22), the ZF equalizer for $\mathbf{s}_k(n)$ can be described by the matrix

$$\boldsymbol{\Gamma}_k^{(zf)} = \mathbf{F}^{-1}\tilde{\mathbf{H}}_k^{\mathcal{H}} \tag{23}$$

and thus the decision vector $\mathbf{z}_k(n) := [z_{k,1}(n), z_{k,2}(n)]^T = \boldsymbol{\Gamma}_k^{(zf)}\tilde{\mathbf{y}}_k(n)$ is given by

$$\mathbf{z}_k(n) = A(|H_1(\rho_k)|^2 + |H_2(\rho_k)|^2)\mathbf{s}_k(n) + \boldsymbol{\eta}_k(n), \tag{24}$$

where $\boldsymbol{\eta}_k(n) := [\eta_{k,1}(n),\ \eta_{k,2}(n)]^T = \mathbf{F}^{-1}\tilde{\mathbf{H}}_k^{\mathcal{H}}\tilde{\mathbf{w}}_k(n)$. Because $w(n)$ is white with power spectral density N_0 and $\tilde{\mathbf{H}}_k$ is unitary, it follows that $\boldsymbol{\eta}_k(n)$ remains white with covariance matrix

$$R_{\eta_k} = (|H_1(\rho_k)|^2 + |H_2(\rho_k)|^2)\mathbf{v}^{\mathcal{H}}(\rho_k, P)\mathbf{v}(\rho_k, P)\frac{N_0}{2}\mathbf{F}^{-2}\mathbf{I}. \tag{25}$$

Now two remarks are in order: i) since $\boldsymbol{\eta}_k(n)$ is white, the detection of $\mathbf{s}_k(n)$ amounts to solving two single symbol detection problems separately without loss of performance, and since $\mathbf{H}_k$ is unitary, ZF equalization does not enhance the noise; ii) thanks to our design, the detection of $2K$-long data blocks has been cast into $2K$ independent single symbol detection problems. Thus, the complexity of our receiver is proportional to the block length K.

We derive now the BER assuming a QPSK modulation scheme. Our figure of merit is the average BER, defined as $\bar{P}_e = (2K)^{-1}\sum_{k=1}^{K}(\bar{P}_{1,k} + \bar{P}_{2,k})$, where $\bar{P}_{1,k}$ and $\bar{P}_{2,k}$ denote the BERs for the sequences $s(2nK + k - 1)$ and $s((2n+1)K + k - 1)$. It follows from (24) and (25) that

$$\bar{P}_e = \frac{1}{2K}\sum_{k=1}^{K}\sum_{i=1}^{2}\mathcal{Q}\left(\sqrt{\frac{A^2E_bd_i^2(|H_1(\rho_k)|^2 + |H_2(\rho_k)|^2)}{N_0\mathbf{v}^{\mathcal{H}}(\rho_k, P)\mathbf{v}(\rho_k, P)}}\right), \tag{26}$$

where $\mathcal{Q}(\cdot)$ denotes the $\mathcal{Q}$-function. Note that because each symbol is transmitted twice, we divide the transmit power by two for each transmit antenna to obtain (26). When points $\{\rho_k\}_{k=1}^{K}$ are equispaced around the unit circle as in (12), (26) can be further simplified to

$$\bar{P}_e = \frac{1}{2K}\sum_{k=1}^{K}\sum_{i=1}^{2}\mathcal{Q}\left(\sqrt{\frac{Kd_i^2(|H_1(\rho_k)|^2 + |H_2(\rho_k)|^2)E_b}{(K+L)N_0}}\right). \tag{27}$$

5 Simulations

We have also tested the performance of our design by numerical simulations. In order to constrain the total transmit power, we structured the precoder $\mathbf{F}$ as

$$\mathbf{F} = \begin{bmatrix} \sqrt{2-\alpha^2} & 0 \\ 0 & \alpha \end{bmatrix} \tag{28}$$

where α is a design parameter belonging to the interval $(0, \sqrt{2})$. In all simulations, the channels were assumed to be Raleigh faded and $\{\rho_k\}_{k=1}^{K}$ were chosen equispaced around the unit circle.

Example 1: We compare the BER of our proposed transceiver with conventional OFDM systems, assuming frequency-selective channels of orders 1 and 2. The results are shown in Figs. 3 and 4. For OFDM systems, 32 subcarriers are used and the channels are assumed known at the receiver. For our scheme, we assume $\alpha = 0.6$, $K = 16, L = 1$ (Fig. 3) and $K = 32, L = 2$ (Fig. 4). The figures show the BER obtained with OFDM (dotted line) and with our method assuming: i) perfectly known channels (dashed line); ii) blind channel estimation (solid line). The solid line with the plus reports the theoretical results obtained from (27). The blind channel estimate has been obtained by using only 32 blocks of data (i.e., 64 symbols). From Figs. 3 and 4, we infer that our design gives superior performance over conventional OFDM, even when the channels are estimated blindly.

The conventional OFDM transmits each symbol through a single fading channel. In contrast, our proposed system transmits each symbol twice through two different fading channels. Equation (24) shows that the equivalent channel gain for each symbol is the sum of the squares of two different channels which is more reliable than a single channel. This explains why the proposed system outperforms OFDM.

Example 2: To investigate the effect of α in (28), we evaluated the BER and the MSE of the channel estimates as a function of α. In general, α cannot be too close neither to 1, where we have ambiguity problems, nor to 0 where one channel experiences too low SNR. In fact, looking at the results shown in Figs. 5 and 6 obtained with $E_b/N_0 = 16$ dB, we see that there is indeed an optimal value for α. For this example, the α that minimizes the BER is around 0.6.

6 Conclusions and Discussion

In this paper we designed a novel space-time transceiver suitable for multipath channels. Relying on symbol blocking, space-time block codes designed for flat fading channels were extended to frequency-selective channels. A blind channel estimation algorithm was also developed by exploiting the specific structure of space-time block codes. In addition to the simplicity of the proposed receiver, numerical simulations demonstrated the system's superior performance. Future research topics include optimal $\{\rho_k\}_{k=1}^{K}$ allocation algorithms when the channels are known at the transmitter, extension to more than two transmit antennas and multiple receive antennas, blind channel estimation without requiring precoder $\mathbf{F}$ [6], and multiuser extensions.

References

[1] D. Agrawal, V. Tarokh, A. Naguib, N. Seshadri, "Space-Times Coded for High Data Rate Wireless Communication over wideband channels," *Proc. of 48th IEEE Vehicular Technology Conf.*, pp. 2232-2236, Ottawa, Canada, May 1998.

[2] S. M. Alamouti, "A Simple Transmit Diversity Technique for Wireless Communications", *IEEE Journal on Selected Areas in Communications*, vol. 16, no. 8, pp. 1451-1458, October 1998.

[3] Y. Li, N. Seshadri, S. Ariyavisitakul, "Channel Estimation for OFDM Systems with Transmitter Diversity in Mobile Wireless Channels," *IEEE Journal on Selected Areas in Communications*, vol.17, no.3, pp. 461-471, March 1999.

[4] Y. Li, J. Chuang, N. R. Sollenberger, "Transmitter Diversity for OFDM Systems and Its Impacts on High-rate Wireless Networks," *Proc. of 1999 IEEE Int. Conf. on Comm.*, pp. 534-538, Vancouver, Canada, June 1999.

[5] Z. Liu, A. Scaglione, S. Barbarossa, G. B. Giannakis, "Transmit-antennae space-time block coding for generalized OFDM in the presence of unknown multipath," *IEEE Journal on Selected Areas in Communications*, September 1999 (submitted).

[6] Z. Liu, G. B. Giannakis, A. Scaglione, S. Barbarossa, "Decoding and equalization of unknown multipath channels based on block precoding and transmit-antenna di-

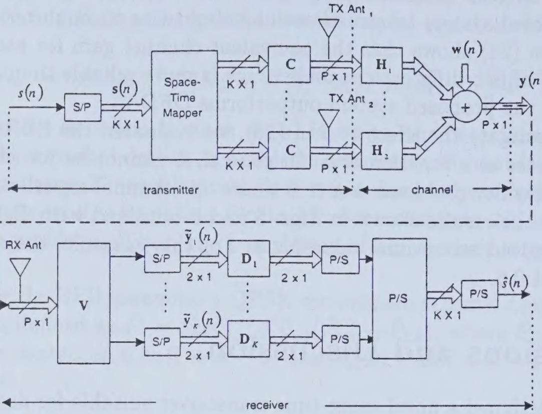

TX Ant 1
w(n)
s(n)
S/P
s(n)
K X 1
Space-Time Mapper
K X 1
C
P x 1
H1
y(n)
+
P x 1
TX Ant 2
K X 1
C
P x 1
H2
transmitter
channel
RX Ant
P x 1
V
ỹ1(n)
S/P
2 x 1
D1
2 x 1
P/S
P/S
K X 1
P/S
ŝ(n)
ỹK(n)
S/P
2 x 1
DK
2 x 1
P/S
receiver

Fig. 1: Chip-rate sampled discrete-time equivalent baseband system model

versity," *Proc. of 33rd Annual Asilomar Conference on Signals, Systems and Computers*, Pacific Grove, CA, Nov. 1999 (to appear).

[7] A. F. Naguib and N. Seshadri and A. R. Calderbank, "Applications of Space-Time Block Codes and Interference Suppression For High Capacity and High Data Rate Wireless Systems," *Proc. of 32nd Annual Asilomar Conference on Signals, Systems and Computers*, pp. 1803-1810, Pacific Grove, CA, 1998.

[8] A. Scaglione and G. B. Giannakis, "Design of user codes in QS-CDMA Systems for MUI Elimination in Unknown Multipath," *IEEE Communications Letters*, vol. 3, no. 2, pp. 25-27, Feb. 1999.

[9] V. Tarokh, N. Seshadri and A. R. Calderbank, "Space-Times Codes for High Data Rate Wireless Communication: Performance Criterion and Code Construction," *IEEE Transactions Theory*, vol. 44, no. 2, pp. 744-765, March 1998.

[10] V. Tarokh, H. Jafarkhani, A. R. Calderbank, "Space-Time Block Codes From Orthogonal Designs," *IEEE Trans. on Information Theory*, vol. 45, no. 5, pp. 1456-1467, July 1999.

[11] A. J. van der Veen, A. Paulraj, "An Analytical Constant Modulus Algorithm," *IEEE Trans. on Siganl Processing*, vol. 44, no. 5, pp. 1136-1155, May 1996.

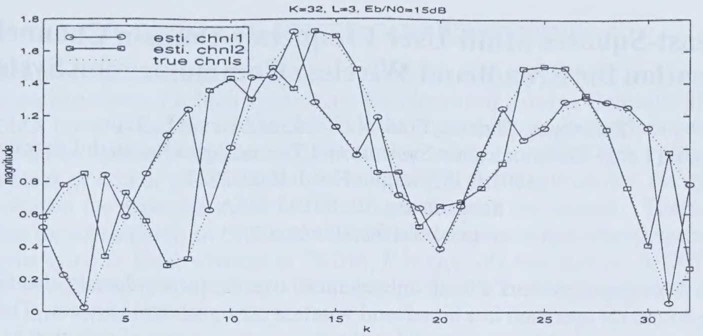


Fig. 2: Estimates of channel magnitude using "deterministic CMA"

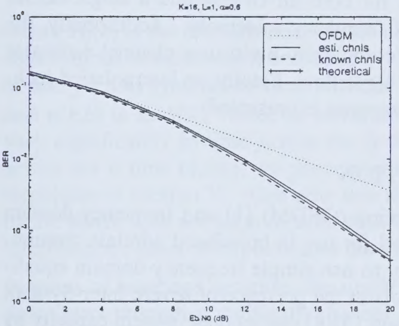


Fig. 3: Known and estimated channels (K=16, L=1)

Fig. 4: Known and estimated channels (K=32, L=2)

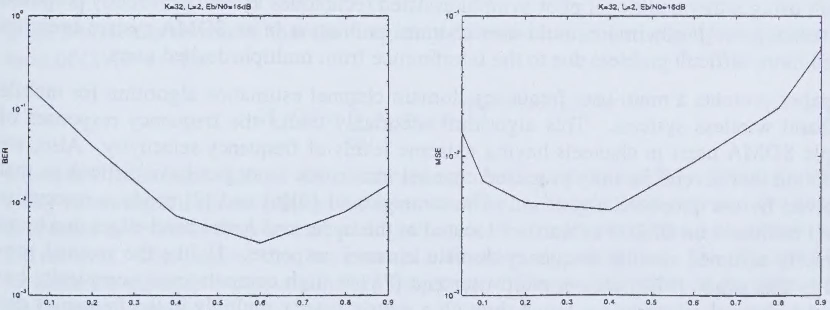


Fig. 5: BER vs. α (E_b/N_0=16dB)

Fig. 6: MSE vs. α (E_b/N_0=16dB)

Least-Squares Multi-User Frequency-Domain Channel Estimation for Broadband Wireless Communication Systems

Timothy A. Thomas, Fred W. Vook, and Kevin L. Baum
Motorola Labs - Communication Systems and Technologies Research Laboratory
1301 E. Algonquin Road, IL02-2928
Schaumburg, IL 60196 USA
tathomas@ccrl.mot.com

Abstract -- This paper presents a least-squares multi-user frequency-domain channel identification algorithm for operation in a broadband wireless communication system. The channel estimation is accomplished by parameterizing the frequency response of each user as the DFT of an unknown impulse response having tap delays uniformly spaced in time. This enables the frequency selectivity of a high-delay spread broadband-channel to be accurately tracked, which is critical for interference suppression and equalization with an adaptive array. The performance of the channel estimator is examined for both an OFDM and a single-carrier system in an environment having extreme levels of frequency selectivity. Additionally, the channel estimator is compared to an existing low complexity multi-user channel estimator and a modified version of the same low complexity algorithm. Finally, an interpolation technique for tracking time variations in the frequency response is presented.

I. INTRODUCTION

Recently Orthogonal Frequency Division Multiplexing (OFDM) [1] and frequency-domain equalization of single carrier [2] have been proposed for use in broadband wireless communication systems. These techniques offer the ability to use simple frequency-domain equalization techniques to recover transmitted data symbols in the presence of severe Inter-Symbol Interference (ISI). In addition, adaptive antenna arrays [3][4] can increase system capacity by allowing multiple users to share the same time-frequency resources, a practice called Spatial Division Multiple Access (SDMA).

Accurate multi-user channel estimation in a broadband SDMA system is critical for both coherent detection as well as interference suppression. In many high-delay spread environments, accurate channel estimation under extreme levels of frequency selectivity is especially difficult using either traditional pilot-symbol assisted techniques as well as recently proposed blind techniques. Furthermore, multi-user channel estimation in an SDMA system becomes an even more difficult problem due to the interference from multiple desired users.

This paper presents a multi-user frequency-domain channel estimation algorithm for mobile broadband wireless systems. This algorithm accurately tracks the frequency responses of multiple SDMA users in channels having extreme levels of frequency selectivity. Also, we have found that several recently proposed channel estimation strategies have difficulties that are solved by our proposed algorithm. The strategies of [5][6] and [7] produce inaccurate channel estimates on OFDM subcarriers located at the upper and lower band-edges due to an incorrectly assumed circular frequency-domain channel response. Unlike the method proposed in this paper, [6][7] are not multi-user and [7] has high computational complexity because the channel estimates are found through a matrix-matrix multiply in the frequency domain. This paper also presents a channel interpolation technique for tracking Doppler-induced variations in the frequency response across the time dimension. This interpolator can be applied to existing channel estimators such as the least squares algorithm proposed in this paper.

II. OFDM AND SINGLE CARRIER OVERVIEW

This section describes the assumed system models for both OFDM [1] and single carrier. The common parameters for both systems are the number of antenna elements, M, the number of SDMA users, J, and the number of unknown (asynchronous or synchronous) users, U. The single carrier system is similar to [2] with there being an OFDM-type cyclic prefix before each block of data or pilot samples. In both OFDM and single carrier, the cyclic prefix must encompass the expected maximum multipath delay of the channel. There are K frequency bins (or subcarriers), an FFT size of N, and B time blocks in a time-frequency slot. In single carrier systems, $K=N$; whereas in OFDM, K is typically less than N. In OFDM, a time block will be called an OFDM baud, which contains the IFFT of the frequency-domain symbols at one time block plus the cyclic prefix.

In the frequency domain, the $M\times1$ received waveform at subcarrier k and baud b, $\mathbf{Y}(k,b)$, is:

$$\mathbf{Y}(k,b)=\sum_{j=1}^{J}\mathbf{H}_j(k)x_j(k,b)+\sum_{u=1}^{U}\mathbf{Y}_u(k,b)+\mathbf{n}(k,b) \tag{1}$$

where $\mathbf{H}_j(k)$ is the $M\times1$ channel vector of user j at subcarrier k. In OFDM, $x_j(k,b)$ is user j's data symbol at subcarrier k and baud b. In single carrier, $x_j(1,b)$ through $x_j(K,b)$ are the FFT of user j's data symbols on time block b. $\mathbf{Y}_u(k,b)$ is the unknown interference for interferer u, and $\mathbf{n}(k,b)$ is an $M\times1$ vector of additive noise. Note that (1) assumes the channel does not vary significantly in time across the B time blocks. When the channel varies significantly across the B time blocks, we propose using pilot blocks spaced in time with the channel interpolator of Section V. Also note that if the unknown interference is asynchronous (i.e., it's cyclic prefix does not align in time with the desired user's cyclic prefixes), then the unknown interference cannot be expressed as the multiplication of a data symbol by a channel vector.

In terms of baseband notation, the single carrier or OFDM time-domain waveform for user j propagates through a multipath channel with P rays arriving at an antenna array of M elements. In equation form, the time-domain impulse response is expressed as:

$$\mathbf{h}_j(t)=\sum_{\ell=1}^{P}\gamma_{j,\ell}\mathbf{a}_{j,\ell}\delta(t-\tau_{j,\ell}) \tag{2}$$

where $\gamma_{j,\ell}$ is the complex gain, $\mathbf{a}_{j,\ell}$ is the array manifold vector [3], and $\tau_{j,\ell}$ is the time arrival for user j's ℓ^{th} multipath ray. ($\mathbf{h}_j(t)$ is $M\times1$.)

In OFDM, it can be shown that the frequency-domain channel is the sampled Fourier Transform of (2) ($k=1\ldots K$):

$$\mathbf{H}_j(k)=\sum_{\ell=1}^{P}\gamma_{j,\ell}\mathbf{a}_{j,\ell}e^{-j2\pi BW(k-1-K/2)\tau_{j,\ell}/K} \tag{3}$$

where BW is the bandwidth of the K subcarriers. An implication of (3) is that there will generally be no significant correlation between the channel at subcarrier $k=1$ and the channel at subcarrier $k=K$ unless $K=N$ and all multipath times of arrival correspond with sample times at the receiver. This implies that any channel estimation algorithm (e.g., [5], [6], and [7]) that assumes the channel "wraps around" in frequency will have difficulties at the edge subcarriers in true channels.

The expression for the single carrier channel is more complex than OFDM since it depends on the assumed pulse waveform. In single carrier with symbol-spaced sampling, the time-domain channel to be estimated is the real RF multipath channel convolved with the pulse symbol waveform truncated to encompass only the significant pulse waveform tails. In Section VII of this paper, we use a pulse waveform with the square-root raised cosine spectrum (β=0.35). It can be shown that using this pulse waveform (or one similar), using a cyclic prefix before each pilot or data block, and sampling the received signal at the symbol rate, then the channel will indeed "wrap around" in the frequency domain. Thus, in a single carrier system, the low complexity multi-user channel estimation procedure described in [5] (for example) will not have problems at the edge subcarriers as it has in OFDM.

III. MULTI-USER FREQUENCY DOMAIN CHANNEL ESTIMATION

The idea behind the new approach is to model each SDMA user's frequency-domain channel as the DFT of L unknown sample-spaced time-domain tap gains, where L is chosen to encompass the maximum expected multipath delay spread. The unknown time-domain tap gains are found through a least-squares matching between each user's transmitted pilot symbols and the received pilot symbols. For ease of presentation, the derivation in this section assumes one continuous training block (i.e., in OFDM one entire baud) and can easily extended to an arbitrary pilot format if desired. In equation form, user j's channel on subcarrier k and antenna m is modeled as:

$$H_{m,j}(k)=\sum_{n=0}^{L-1} h_{m,j}(n)e^{-i2\pi kn/N} \tag{4}$$

where N is the DFT size of the model (which can be chosen independently from the FFT size of OFDM), and $h_{m,j}(n)$ is user j's channel tap on antenna m at time n. For single carrier, N=K, but for OFDM, N must be chosen to be larger than K so that the channel estimates around subcarrier k=1 are not significantly correlated with the channel estimates at subcarriers around k=K as discussed in Section II. A different $H_{m,j}(k)$ is found on each antenna for user j and then stacked in the vector $\mathbf{H}_j(k)$ to get the steering vector at subcarrier k.

Assuming the pilot block is at time b_p, the estimate for the unknown channel time taps on the m^{th} antenna is found by solving the following minimization problem:

$$\min_{\substack{h_{m,\ell}(n)\\ \ell=1\ldots J\\ n=0\ldots L-1}} \sum_{k=1}^{K}\left|\sum_{i=1}^{J} x_i(k,b_p)\sum_{n=0}^{L-1} h_{m,i}(n)e^{-j2\pi kn/N}-Y_m(k,b_p)\right|^2 \tag{5}$$

The estimation problem solves for the $h_{m,\ell}(n)$'s by matching in a least squares sense the known transmitted pilot symbols to the received data. The problem can be reformulated in a matrix equation as follows:

$$\min_{\mathbf{h}_m}\left|\mathbf{Q}\mathbf{h}_m-\boldsymbol{\Psi}_m\right|^2 \tag{6}$$

where: $\mathbf{h}_m=[h_{m,1}(0),\ldots,h_{m,1}(L-1),h_{m,2}(0),\ldots,h_{m,2}(L-1),\ldots,h_{m,J}(0),\ldots,h_{m,J}(L-1)]^T$,

$$\boldsymbol{\Psi}_m=\begin{bmatrix}Y_m(1,b_p)\\ \vdots\\ Y_m(K,b_p)\end{bmatrix}\quad \mathbf{Q}=\begin{bmatrix}\mathbf{q}^H(1)\mathbf{X}(1,b_p)\\ \vdots\\ \mathbf{q}^H(K)\mathbf{X}(K,b_p)\end{bmatrix} \tag{7}$$

$$\mathbf{q}(k)=\left.\begin{bmatrix}\mathbf{f}(k)\\ \mathbf{f}(k)\\ \vdots\\ \mathbf{f}(k)\end{bmatrix}\right\}J\ \mathbf{f}(k)\text{'s},\quad \mathbf{f}(k)=\begin{bmatrix}1\\ e^{j2\pi(k-1)/N}\\ \vdots\\ e^{j2\pi(L-1)(k-1)/N}\end{bmatrix} \tag{8}$$

$$\mathbf{X}(k,b) = diag\{x_1(k,b)\mathbf{1}^T, x_2(k,b)\mathbf{1}^T, \ldots, x_J(k,b)\mathbf{1}^T\} \tag{9}$$

In these equations, $\mathbf{h}_m$ is $JL\times 1$, $\mathbf{Q}$ is $K\times JL$ $\mathbf{q}(k)$ is $JL\times 1$, $\mathbf{f}(k)$ is $L\times 1$, $\mathbf{X}(k,b)$ is $JL\times JL$, Ψ_m is a $K\times 1$ matrix of received pilot data, and $\mathbf{1}$ is a $L\times 1$ vector of ones.

The solution to this minimization problem is:

$$\mathbf{h}_m = \left(\mathbf{Q}^H\mathbf{Q}\right)^{-1}\mathbf{Q}^H\Psi_m \tag{10}$$

Once these time taps are found for each user, they can be converted back to the frequency-domain by (4) (or by efficient FFT processing). Note that if $\Psi = [\Psi_1, \Psi_2, \ldots, \Psi_M]$ $(K\times M)$ and $\mathbf{h}=[\mathbf{h}_1,\mathbf{h}_2,\ldots\mathbf{h}_M]$, $(LJ\times M)$ then the solution can be written as:

$$\mathbf{h} = \mathbf{T}\Psi \tag{11}$$

where $\mathbf{T} = \left(\mathbf{Q}^H\mathbf{Q}\right)^{-1}\mathbf{Q}^H$ $JL\times K$.

In some applications, it is helpful to minimize the norm of the time-domain channel taps by diagonally loading $(\mathbf{Q}^H\mathbf{Q})^{-1}$. For example, for multi-user channel estimation in OFDM, the diagonal loading prevents serious channel estimation errors at the edge subcarriers.

IV. REDUCED COMPLEXITY ESTIMATION

The above least-squares channel estimation method requires $MJLK$ complex multiplies plus MJ N-point FFTs. The multi-user channel estimation technique of [5] has significantly fewer computations, and for single carrier is actually equivalent to the above least-squares formulation. Unfortunately in OFDM, the method of [5] produces inaccurate estimates at the edge subcarriers (around k=1 and k=K), which causes a BER floor. To solve the edge subcarrier problem in OFDM, a modified estimator is presented below that combines the least-squares estimator with the reduced complexity estimator of [5].

The reduced complexity channel estimator of [5] works by using a specially designed training sequence. For user i, the training sequence is:

$$x_i(k,b_p) = b_T(k)e^{-j2\pi(i-1)kL/K} \tag{12}$$

where $b_T(k)$ is some constant modulus frequency-domain training sequence (e.g., QPSK), and L is the length of the time-domain channel to be estimated. The time-domain channel estimates for each user are found as follows. First a K-point IFFT is performed on the multiplication of the received pilot symbols on each antenna by the conjugate of the training symbols. In equation form this is:

$$\mathbf{Z} = \mathrm{IFFT}(\Psi \otimes \mathbf{B}_T) \tag{13}$$

where $\mathbf{Z}$ is a $K\times M$ matrix of the time-domain channel estimates of each user as described below, the K-point IFFT is performed column wise, $\otimes$ means an element by element multiply, and $\mathbf{B}_T$ $(K\times M)$ is:

$$\mathbf{B}_T = \begin{bmatrix} b_T^*(1) & \cdots & b_T^*(1) \\ \vdots & & \vdots \\ b_T^*(K) & \cdots & b_T^*(K) \end{bmatrix} \tag{14}$$

Because of the linear phase term on each user's training sequence, user j's time-domain channel estimate on antenna m is simply rows $(j$-1$)L$+1 through jL of column m of $\mathbf{Z}$. Thus the computation complexity of this algorithm requires KM complex multiplies plus M K-point IFFTs and MJ K-point FFTs.

For a single-carrier system, the reduced complexity estimator of [5] works well as is. However, the following modification can be employed in OFDM to correct for the poor performance at the edge subcarriers. First apply the reduced complexity algorithm of [5] and throw away the first and last K_B (K_B<<K) subcarriers. Then use the least squares approach of Section III with L_B time taps (L_B<<L) and FFT size N_B (N_B>K_B) on the first and last K_B subcarriers separately. In other words, the modified reduced complexity algorithm simply does a least-squares fitting on the first and last K_B subcarriers and uses the reduced complexity channel estimates elsewhere.

The modified reduced complexity algorithm requires $2JMLL_BK_B+KM$ complex multiplies plus $2JM$ N_B-point FFTs plus M K-point IFFTs plus JM K-point FFTs. For comparison, consider an OFDM system with M=8 antennas, J=3 users, K=256 subcarriers, N=512, L=64, L_B=8, K_B=32, and N_B=64. The number of complex multiplies for an N-point FFT or IFFT is assumed to be $(N/2)\log_2(N)$ (which may be lower in practice). The least-squares approach requires 448,512 complex multiplies. The reduced complexity channel estimator of [5] requires 34,816 complex multiplies (13 times less than least squares), and the modified reduced complexity method of this section requires 56,320 complex multiplies (8 times less than least squares and 1.6 times more than the reduced complexity algorithm of [5].

V. CHANNEL INTERPOLATOR FOR TIME-VARYING CHANNELS

The channel estimators discussed so far assume a constant channel over the block of received pilot symbols. These estimators work well only at data symbols where the channel has not significantly changed from the pilot symbols locations. In this section we discuss a means of interpolating in time between blocks of channel estimates at different times in order to find channel estimates at all times of interest. The goal is to take the existing channel estimates found at different time blocks, b_p (p=1...P) and interpolate / extrapolate these estimates to estimate the channel response everywhere. The derivation assumes knowledge of the channel at time blocks b_1 through b_P, and is described in terms of the frequency-domain channel $\mathbf{H}(k,b)$. (This derivation is easily extended to the time-domain channel corresponding to a single lag value, as is done for Figure 5 of Section VII.)

The time-varying channel is found at all times by parameterizing the time-varying $H_{j,m}(k,b)$ of user j on antenna m by V_T=(2V+1) time-invariant "Doppler" channels, $g_{j,m,v}(k)$ (v=-V,...,+V), as given in the following equation:

$$H_{j,m}(k,b)=\sum_{v=-V}^{+V} g_{j,m,v}(k)e^{j2\pi vb/N_k}=\mathbf{c}^H(b)\mathbf{g}_{j,m}(k) \tag{15}$$

where N_k is chosen so that the frequency associated with the "Doppler" channel at v=V corresponds to the maximum expected Doppler frequency and $\mathbf{c}(b)$ and $\mathbf{g}_{j,m}(k)$ are $V_T\times 1$ vectors defined as:

$$\mathbf{c}(b)=\begin{bmatrix} e^{j2\pi Vb/N_k} \\ \vdots \\ e^{-j2\pi Vb/N_k} \end{bmatrix},\quad \mathbf{g}_{j,m}(k)=\begin{bmatrix} g_{j,m,-V}(k) \\ \vdots \\ g_{j,m,+V}(k) \end{bmatrix} \tag{16}$$

The "Doppler" channels, $g_{j,m,v}(k)$, are found by solving the following equation in the least squares sense.

$$\min_{\mathbf{g}_{j,m}(k)} \left|\mathbf{C}\mathbf{g}_{j,m}(k) - \boldsymbol{\Theta}_{j,m}(k)\right|^2 \quad (17)$$

where $\mathbf{C}$ ($P{\times}V_T$) and $\boldsymbol{\Theta}_{j,m}(k)$ ($P{\times}1$) are:

$$\mathbf{C} = \begin{bmatrix} \mathbf{c}^H(b_1) \\ \vdots \\ \mathbf{c}^H(b_P) \end{bmatrix}, \quad \boldsymbol{\Theta}_{j,m}(k) = \begin{bmatrix} H_{j,m}(k,b_1) \\ \vdots \\ H_{j,m}(k,b_P) \end{bmatrix} \quad (18)$$

The solution to (17) is:

$$\mathbf{g}_{j,m}(k) = \left(\mathbf{C}^H\mathbf{C}\right)^{-1}\mathbf{C}^H\boldsymbol{\Theta}_{j,m}(k) \quad (19)$$

Once the "Doppler" channels are found, the channel at any arbitrary time *b* is found through (15).

VI. ANTENNA ARRAY COMBINING METHODS

After the channels for all users have been estimated, these estimates are used in antenna combining algorithms [3][4] to estimate the transmitted symbols. Let the output of the combining operation be:

$$r_j(k,b) = \mathbf{w}_j^H(k)\mathbf{Y}(k,b) \quad (20)$$

In OFDM, $r_j(k,b)$ is the estimate of user *j*'s symbol at time block *b* and subcarrier *k*. In a single carrier communication system with cyclic prefixes on the data time blocks, the set of $K=N$ time domain symbol estimates for user *j* and time block *b* is the IFFT of $\{r_j(1,b),\ldots r_j(K,b)\}$. Note that these combining weights do not assume time variations. With a time varying channel estimator, such as the one in Section V, a different set of weights can be found on each time block *b*.

The choice of combining algorithm depends on whether or not significant unknown (i.e., non-SDMA) interference is present. The first method for computing the weights assumes there is no unknown interference and is called the Partial Reconstruction method. The term "Reconstruction" refers to the technique of "reconstructing" the optimal combining equations [3] using the channel response estimates of the known users. The weight for user *j* on subcarrier *k* is given by:

$$\mathbf{w}_j(k) = \left\{\sum_{i=1}^{J}\mathbf{H}_i(k)\mathbf{H}_i^H(k) + \sigma_n^2\mathbf{I}\right\}^{-1}\mathbf{H}_j(k) \quad (21)$$

where σ_n^2 is the estimated noise power on the array.

When unknown interference is significant, an estimate of the unknown interference plus noise signals can be found at the pilot symbol locations. This is done by subtracting out each SDMA users' signal contribution from the received signal vector at each pilot symbol by using their estimated steering vectors and the known pilot symbols. What is left is an estimate of the received unknown interference plus noise at each pilot symbol. This remaining signal is $\mathbf{x}_u(k,b_p)$ and has the following form:

$$\mathbf{x}_u(k,b_p) = \mathbf{Y}(k,b_p) - \sum_{j=1}^{J}\mathbf{H}_j(k)x_j(k,b_p) \quad (22)$$

The external interference plus noise spatial correlation matrix, $\mathbf{R}_u(k)$ can then be found by averaging (22) over some number of pilot symbol locations:

$$\mathbf{R}_u(k) = average(\mathbf{x}_u(k,b_p)\mathbf{x}_u^H(k,b_p)) \tag{23}$$

where average means to average what is inside the parentheses over some number of subcarriers and time blocks. (In the simulations, the averaging is done over 16 consecutive subcarriers). This method is referred to as the Full Reconstruction method and the weight for user *j* on subcarrier *k* is given by:

$$\mathbf{w}_j(k) = \left\{\sum_{i=1}^{J}\mathbf{H}_i(k)\mathbf{H}_i^H(k) + \mathbf{R}_u(k)\right\}^{-1}\mathbf{H}_j(k) \tag{24}$$

Note that $\mathbf{R}_u$ is written as a function of *k* to refer to the fact that $\mathbf{R}_u$ may be computed to vary (possibly in a block fashion) across subcarriers in order to track the frequency variations of the unknown interference. Also note that accurate channel estimates, such as those given by the channel estimation procedure in Section III, are needed for the Full Reconstruction method to work well. Without accurate estimates, part of the desired signal might be suppressed resulting in poor performance as can happen in OFDM with the channel estimates from [5].

The final set of weights, used for comparison purposes, are referred to as theoretical optimal combining weights. These weights define a performance bound to compare with the proposed algorithms. These weights assume perfect knowledge of the frequency-domain waveform of all *U* unknown interferers, the frequency-domain channel of the desired users, the signal statistics of the desired users, and the noise statistics. The optimal combining weights at subcarrier *k* for user *j* are given as:

$$\mathbf{w}_j(k) = \left(\sum_{j=1}^{J}\mathbf{H}_j(k)\mathbf{H}_j^H(k) + \sum_{u=1}^{U}\mathbf{Y}_u(k)\mathbf{Y}_u^H(k) + \sigma_n^2\mathbf{I}\right)^{-1}\mathbf{H}_j(k) \tag{25}$$

The exact frequency-domain signals of the unknown interferers are used in (25) instead of their frequency-domain channels because the interference is assumed to be asynchronous. Asynchronous interference cannot be modeled in the frequency domain as the multiplication of a data symbol by the channel. This is because the *N* time-domain samples that are FFT'd by the receiver will contain multipath components arriving from both the previous and current time-blocks of the asynchronous interferer.

VII. SIMULATION RESULTS

This section presents a comparison of the different channel estimation methods for a single-carrier system and an OFDM system. Some common system parameters are an M=8 element antenna array, SDMA=3 16-QAM users, and a 6.0 MHz bandwidth. (In OFDM, this bandwidth encompasses the *K* subcarriers. In single carrier, this bandwidth corresponds to the total width of the frequency-domain pulse waveform). The channels for the first set of simulation runs (Figures 1 through 4) have no time variations whereas the channels for the second set of simulation runs (Figures 5 and 6) have varying degrees of time variations. For the Full Reconstruction combining weights, the averaging of $\mathbf{R}_u$ is done on a disjoint (i.e., non-sliding) block basis where the block size is 16 subcarriers.

For the OFDM system, there are B=3 bauds with the second baud, b_p=2, being all pilot symbols. Also to improve the performance of the multi-user least squares estimator near the edge frequencies, there are pilots on subcarriers k=1 and k=K on all three bauds, and diagonal loading with 0.01**I** was used in (10). The FFT size is N=512, the number of usable subcarriers is K=448, and the cyclic prefix length is 64 which corresponds to 9.33 μsec. For all OFDM channel estimation algorithms, L is chosen to be 57 which corresponds to a maximum multipath delay of 8.2 sec. For the modified reduced complexity algorithm, L_B=8, N_B=64, K_B=32, and only 25 out of the 32 (k=1 through 25 and k=K-24 through K) least squares channel estimates are kept because of the channel estimation errors that occur at the subcarriers on the edge of the blocks of 32 subcarriers.

For single carrier, there are N=K=512 pilots with blocks of 1024 data symbols on both sides of the pilots and the symbol rate is 0.225 μsec. The pulse waveform has the square-root raised cosine spectrum (β=0.35) and the channel is this pulse convolved with the real RF multipath channel. (The estimator assumes three significant "tails" in the pulse waveform.) There is a cyclic prefix before each data and pilot block of length 43 which corresponds to 9.45 μsec. For single carrier, the performance of the least-squares channel estimation is identical to the reduced complexity estimation when the pilot sequences of Section IV is used. Therefore, the results shown for single-carrier are only for the reduced complexity estimator with L=43.

In the legends of the figures, OFDM LS is the least squares estimator for OFDM, OFDM (SC) IFFT/FFT is the reduced complexity estimator for OFDM (Single Carrier), OFDM LS-IFFT/FFT is the modified reduced complexity algorithm for OFDM, and OFDM (Single Carrier) Optimal is the theoretical optimal combining weights (25) for OFDM (Single Carrier).

Figure 1 shows the BER performance of the Partial Reconstruction weights for the different systems and channel estimators when τ_{max}=0 μsec. (The 0 μsec delay spread makes the channel the same at all subcarriers). As can be seen, the performance of each algorithm in each system is virtually identical except for OFDM, where the modified reduced complexity algorithm has degraded performance because of the block least-squares channel estimation being done at the edge subcarriers. However, once there is some delay spread on the channel, the reduced complexity algorithm's performance for OFDM degrades as shown in Figure 2 for τ_{max}=8.0 μsec. Besides the performance hit for the OFDM reduced complexity channel estimators, the frequency diversity benefit of single carrier is also evident.

Figure 3 shows the effects of two unknown interferers (at SIR=15 dB) where the channels for each incident signals have τ_{max}=8.0 sec. The combining weights used for these simulation runs are the Full Reconstruction combining weights (24). As can be seen, the Full Reconstruction method for single carrier and for least-squares in OFDM work well because the channel estimates are more accurate than the reduced complexity OFDM estimates. As before, single carrier has a frequency diversity advantage over OFDM which can be corrected in OFDM through appropriate coding.

Figure 4 shows the effects of the choice of L for the least-squares channel estimator, Partial Reconstruction combining weights, and the OFDM system for a channel having its maximum delayed ray at tap number 33. L must be at least 34 taps long to eliminate the BER floor due to the simulated rays not arriving at sample times. Note that it is better to overestimate L than to underestimate it.

Figures 5 and 6 show simulation results using the channel interpolator of Section V for a single carrier system with the same parameters as given above except for the data/pilot slot format. The data/pilot slot format consists of 4 pilot blocks of 256 symbols plus a cyclic prefix of length 43. In between each pair of pilot blocks there are 8 data blocks of 256 symbols each with a length 43 cyclic prefix (the data burst spans 1.9 msec). This slot format was chosen to be fairly long in time so that the time variations are significant. The FFT size was chosen small enough so that the channel is approximately constant across the N=256 symbols. The channel estimates for each of the three SDMA users on each pilot block are computed with the reduced complexity estimator with L=42 followed by the channel interpolator (operating on the time-domain channel taps) with V_T=3 (V=1) and N_K=19,121 which corresponds to a Doppler frequency of f_d=232 Hz. A different set of Partial Reconstruction combining weights are found for each user on each block of 256 data symbols. Figure 5 shows how well the channel interpolator tracks the direct ray of the true channel of user 1 on antenna 1 when the SNR per one SDMA user is 20 dB. The vertical dash-dot lines indicate the locations of the four pilot blocks. As can be seen, the estimator tracks the user's channel at this lag value extremely well even when the channel experiences a null between pilot symbols.

Finally, Figure 6 shows the BER of the channel estimator for different delay spreads and Doppler frequencies. This is compared to the theoretical optimum tracking for f_d=0 Hz and 8.0 μsec delay spread. As can be seen, the estimator works well for all delay spreads and Doppler spreads and even degrades gracefully at a Doppler frequency (348 Hz) that is larger than the interpolator was designed for (232 Hz).

VIII. CONCLUSION

In this paper, we introduced a least-squares multi-user frequency-domain channel estimation technique that parameterizes each SDMA user's channel as a small number of time taps. This parameterization allows for accurate estimates of each desired users' frequency-domain channels even at the edge subcarriers in OFDM. In addition, two reduced complexity algorithms were discussed that have a large computational savings over the least squares approach. However there is a performance hit when applying the reduced complexity algorithms to an OFDM system. Finally a method for interpolating channel responses in time was presented that allows excellent tracking of time-variations between existing channel estimates.

REFERENCES

[1] J. A. C. Bingham, "Multicarrier Modulation for Data Transmission: An Idea Whose Time Has Come," *IEEE Communications Magazine*, Vol. 28, pages 5-14, May 1990.

[2] M. V. Clark, "Adaptive Frequency-Domain Equalization and Diversity Combining for Broadband Wireless Communications", *IEEE Journal on Selected Areas in Communications*, Vol. 16, pages 1385-1395, October 1998.

[3] R. T. Compton Jr., *Adaptive Antennas: Concepts and Performance*, Englewood Cliff, NJ: Prentice Hall, 1988.

[4] J. H. Winters, "Optimum Combining in Digital Mobile Radio with Cochannel Interference," *IEEE Trans on Vehicular Tech.*, Vol. 33, No. 3, August 1984

[5] B. Steiner, "Time Domain Channel Estimation in Multi-carrier-CDMA Mobile Radio System Concepts", in *Multi-Carrier Spread-Spectrum*, pp. 153-160, Kluwer Academic Publishers, 1997.

[6] J. van de Beek, et al., "On Channel Estimation in OFDM Systems", Proceedings VTC 95, pp. 815-819

[7] Y. (G.) Li, L. J. Cimini Jr. and N. R. Sollenberger, "Robust Channel Estimation for OFDM systems with Rapid Dispersive Fading Channels," *IEEE Transactions on Communications*, vol. 46, pp. 902-915, July 1998.

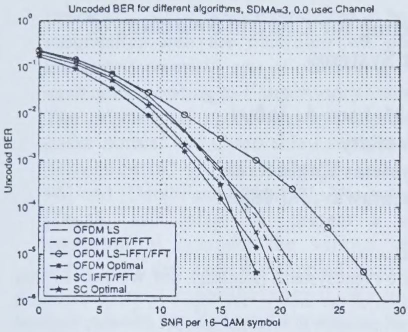


Figure 1. BER Performance Comparison for τ_{max}=0 μsec

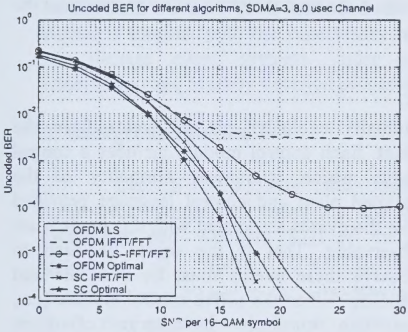


Figure 2. BER Performance Comparison for τ_{max}=8.3 μsec

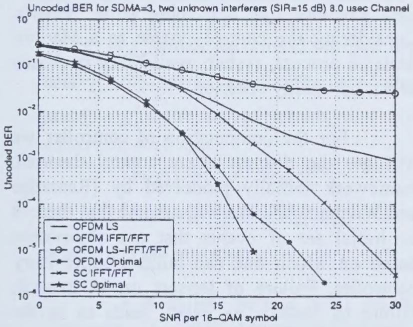


Figure 3. BER Performance Comparison for Two Unknown Interferers (SIR=15 dB)

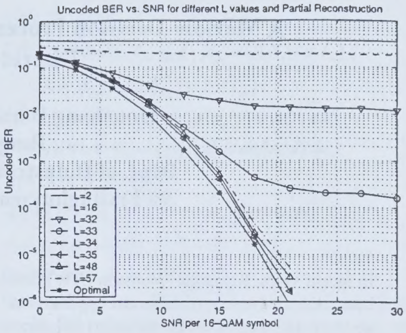


Figure 4. The Effects of Choice of *L* on the Least Squares Estimator and OFDM

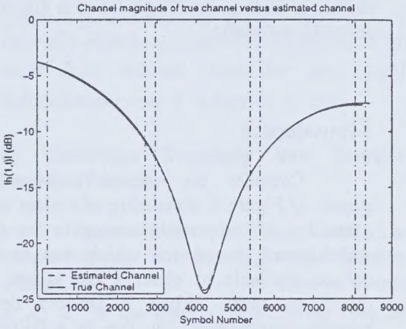


Figure 5. Channel Interpolation Results for Reduced Complexity Estimator

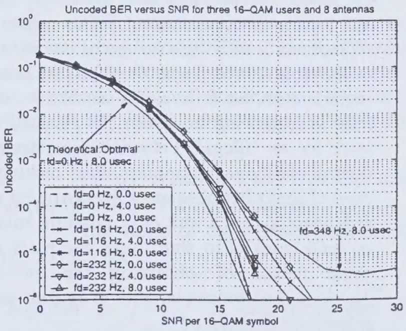


Figure 6. BER for Channel Interpolator for different Doppler Freqs. and Max. Delays

A Markov Decision Process Modeling for Control Switching of Discrete Event Systems

Houshang Darabi and Mohsen A. Jafari
Rutgers University
Dept. of Industrial and Systems Engineering,
96 Frelinghuysen Rd. Piscataway, NJ 08854

Abstract

In recent years, Ramadge-Wonham (RW) theory of discrete event systems [Ramadge and Wonham 87] has been extended to many different applications. An extension of the theory, developed by [Haji-Valizadeh and Loparo 96 and Darabi and Jafari 98], focuses on the computation of minimal number of events required to be observed by the RW supervisory controller. We utilize these results to design a framework for controller switching and reconfiguration when required by the changes in the plant environment (e.g., sensors failures or repairs). We show that the problem can be reduced to a Markov decision process under a number of assumptions. The solution to this model provides a dynamic switching policy to maximize the controller availability.

1-Introduction

Consider the closed loop control system of Figure 1, consisting of a plant and a controller. The events occurring in the plant are observed by sensors, which may or may not be available to observe the events and report on them. When a sensory device becomes unavailable, say due to a failure, it can no longer observe and report the corresponding event. This will continue until the sensor becomes available again, say due to its repair or replacement. The repair or replacement activity obviously takes some measurable time. Depending on the control requirements, the failure of a sensory device or lack of observing an event, in general, may lead to an illegal behavior (in terms of controllability requirements), thus making the current controller obsolete. Obviously, for non-redundant systems, the only possible solution in such cases is to shut down the plant until some means of observing the event is provided, i.e. the unavailable or broken sensor is repaired or replaced.

We consider a number of different standby control modules within the control system, each one requiring observation and reporting of a different but not necessarily mutually exclusive set of events from the plant. Each control module includes a control logic, which guarantees normality of the coupled plant and control language provided that the events associated with this module are observable. That is the sensor required to observe these events must be available and working. When the active control module (i.e., the one currently in charge of plant supervision) becomes non-conforming we switch to another control module. This new control module must be conforming based on the existing set of observable events. Considering the control system as a Markov decision process, system availability is chosen as our optimality criterion.

For the computation of controller we will use the discrete event control theoretic framework initially pioneered by [Ramadge and Wonham 87]. We will assume that there is an initial set of events that are observable, based on which one can build a supervisory controller. Failure of sensors or, in general, failure to observe an event, leads us to the problem of controller synthesis under partial observation, originally discussed by [Lin and Wonham 88] and [Cieslak et al. 88]. They separately present results for synthesis of supervisory controllers under normality condition. Since, we are interested in building redundancy into the controller we need to compute the various sets of observable events, each corresponding to a supervisory controller. The problem of optimizing event projection under normal and observable controllers have been discussed by [Haji-Valizadeh and Loparo 96] and [Darabi and Jafari 98]. The authors in [Darabi and Jafari 98] present a simple algorithm to compute the various sets of projections that keep the coupled language of the plant and controller normal. We will use their results and the results provided by [Vaz and Wonham 86] here to model the controller-switching problem as a Markov decision process. In the next section we provide a background review. It is assumed that the reader is familiar with the supervisory control of discrete event systems under partial observation and Markov decision processes.

2. Background Review

Let $\mathrm{S}=(S,\Phi)$ be the complete supervisor (synthesized using the RW theory [Ramadge and Wonham 87]) of the plant generator G where: $G=(Q,\Sigma,f,q_0,Q)$, $S=(X,\Sigma,g,x_0,X)$ and $\Phi:X\rightarrow\Gamma$. We assume that the event set Σ is finite. Let $K=L(\mathrm{S}/G)$ be the language generated by the coupled system S/G and assume that the supervisor S can observe all the events of Σ.

A *projection* (mask) P is a function $P:\Sigma\rightarrow\Sigma_o$ which partitions Σ to two disjoint sets: Σ_o^P: the set of events that are observed and $\Sigma-\Sigma_o^P$: the set of events that are not observed, such that

$$\begin{cases} P(\sigma)=\sigma & \text{if} \quad \sigma\in\Sigma_o^P \\ P(\sigma)=\varepsilon & \text{if} \quad \sigma\in\Sigma-\Sigma_o^P \end{cases}$$

where ε is the null event.

K is called *normal* with respect to language L and projection P (shown by $K\sim N(L,P)$) if $L\cap P^{-1}P(K)=K$. A *cover* of supervisor $\mathrm{S}=(S,\phi)$ is a family $C=\{X_i,i\in I\}$ of subsets of X with the following properties:

1)$\forall i\ X_i\neq\Phi$;

2)for a subset $I_m\subseteq I$, $X_m=\bigcup\{X_i|\,i\in I_m\}$,

$X-X_m=\bigcup\{X_i|\,i\in I-I_m\}$;

3)$(\forall i,\sigma):(\exists\, y\in X_i)\wedge(g(\sigma,y)\ \textit{is defined})$ $\Rightarrow(\exists j)(\forall x\in X_i)\wedge(g(\sigma,x)\ \textit{is defined})\Rightarrow$ $g(\sigma,x)\in X_j$; *and*

4)$(\forall\, i,\sigma)(\forall x_1,x_2\in X_i):\phi[x_1](\sigma)\neq dc$ $\neq\phi[x_2](\sigma)\Rightarrow\phi[x_1](\sigma)=\phi[x_2](\sigma)$.

In our case, $I_m=I$ since $X_m=X$.

Next we present the definition of legal one step controller switching. We use the concept of reduced supervisor [Vaz and Wonham 86] in this definition. Later we construct the action space of Markov decision process using the possible controller switching options.

One step switching (controller switching) from state X_i of reduced supervisor S to X_j of reduced supervisor S′ is legal if $X_i\subseteq X_j$.

3- Controller Switching and Markov Decision Process

Assume that the supervisor S in coupled system S/G is using single function sensors, Dj, $j=1,2,\cdots,n$, to observe events $e_1,e_2,\ldots e_n$ of the plant G respectively. We let $\Sigma=\{e_1,e_2,\cdots,e_n\}$. It is assumed that under full observation policy, the coupled language is $K=L(\mathrm{S}/G)$. Let $O=\{P_1,P_2,\cdots,P_m\}$ be a subset of all normal projections that can be obtained using the algorithm given by [Darabi and Jafari 98]. Also suppose that it is possible to build a reduced supervisor [Vaz and Wonham 86] with the following properties for each projection:

-All the events in $\Sigma-\Sigma_o^P$ appear as self loop events in the reduced supervisor.

-The states of the reduced supervisor are disjoint sets.

Suppose Dj, $j=1,2,\cdots,n$ can be in one of three states idle(I), working(W) or unavailable (U) at any time. Let $x_j(t)$ denote the state of Dj at time t where $t\geq 0$ and $j=1,2,\cdots,n$. Also let $X(t)=(x_1(t),x_2(t),\cdots,x_n(t))$. Let $I(t)$ be the active controller state at time t. We assume that the controllers' states constructed on the set O, are indexed by integer numbers $1,\cdots,u$. Index 0 is used when "no controller" is used to control the plant. The system S/G is in mode 1 at time t if the following two conditions are held: 1) $X(t)$ supports the events that are expected to be observed by S, i.e. $\forall i : x_i(t) = W$ if event e_i is to be observed by S, and 2) the current state of the supervisor with full observation (if this supervisor was used to control the plant) belongs to the current state of S (we recall that if S is a reduced supervisor then any state of S is an element of a cover). The coupled system is in mode 2 at time t, otherwise. Our goal is to design a controller switching policy, among the controllers constructed using the projections of set O, to maximize the percentage of time that the system acts in its mode 1. Our action space in this case is defined by the atomic actions of turning on/off $((I \to W)/(W \to I))$ of sensory devices Dj, $j = 1,2,\cdots,n$; and legal one step switching between controllers. Let B_{y/X_i} be the probability of being in state $y \in X$ given that the current supervisor, is in state X_i and $y \in X_i$. It is assumed that the probabilities B_{y/X_i}'s are given.

The above problem (designing an optimal controller switching policy) can be formulated by Markov Decision Process (MDP) [Ross 97]. Let $Y(t) = [X(t), I(t)]$ be the state of MDP with time index $t = 0,1,\cdots$. Let $a(t)$ be the action made at time t. The possible values for $a(t)$ depend on $Y(t)$. As we already mentioned, $a(t)$ is defined based on the possible combinations of atomic actions $(I \to W)-(W \to I)$ for Dj, $j = 1,2,\cdots,n$ in state $Y(t)$ and the related controller switching options. We assume that if $I(t) = 0$ then the controller chosen by $a(t)$ (if any) must start from its initial state. The MDP evolution then happens as follow: At time t, the state of the system $Y(t)$, is observed; action $a(t)$ from the set of possible actions in $Y(t)$ is selected and performed. Consequently, the state of the system is changed to $Y'(t) = [X'(t), I'(t)]$. It is assumed that $a(t)$ can be performed instantaneously. While in $Y'(t)$ we look for events that could possibly happen in the next time step $(t, t+1]$. The potential events are the failures or the repair completion of sensory devices and regular events defined by Σ. It is assumed that all such events happen independently. It is also assumed that the time step $(t, t+1]$ is selected such that the probability of occurrence of two or more events of any type in this interval is negligible. It is assumed that time to failure and time for repair of sensory device D_j are exponentially distributed random variables, denoted by F_j and R_j, with parameters f_j and r_j respectively. Also we let $\lambda_{j/X}$ be the parameter of exponentially distributed random variable $H_{j/X}$ denoting the time to occurrence of event $j \in \Sigma$ when it becomes enabled by the control feedback policy of state X. Let $P_{ij}(a)$ denote the probability of transition to state j in the next time step given that the current state is i and action a is selected. In other words:

$$P_{ij}(a) = P\{Y(t+1) = j / Y(t) = i, a = (Y(t) \to Y'(t))\}$$

.

We compute $P_{ij}(a)$ using the following procedure:

1- Compute $P(e_i)$, $e_i \in \Sigma$, where

$$P(e_i) = \begin{cases} 0 & \text{if } I'(t) = 0 \\ \Pr(H_{i/I(t)}) = 1 - e^{-\lambda_{i\ I(t)}} & \text{if } I(t) = 0, I'(t) \neq 0 \\ \sum\limits_{x \in I(t)} B_{x/I(t)} . \Pr(H_{i/I(t)} \leq 1) & \text{if } I(t) \neq 0, I'(t) \neq 0 \\ \quad = \sum\limits_{x \in I(t)} B_{x/I(t)} . (1 - e^{-\lambda_{i.I(t)}}) & \end{cases}$$

2- Compute $P(D_j)$, $1 \leq j \leq n$, where:

$$P(D_j) = \begin{cases} \Pr(F_j \leq 1) = 1 - e^{-f_j} & \text{if } x_j(t) = W \\ \Pr(R_j \leq 1) = 1 - e^{-r_j} & \text{if } x_j(t) = U \\ 0 & \text{if } x_j(t) = I \end{cases}$$

3- Let

$$\overline{P}(e_i) = P(e_i).[\prod_{\substack{k=1 \\ k\neq i}}^{n}(1-P(e_k))]\times$$

$$[\prod_{k=1}^{n}(1-P(D_k))]$$

$$\overline{P}(D_j) = P(D_j).[\prod_{\substack{k=1 \\ k\neq j}}^{n}(1-P(D_k))]\times$$

$$[\prod_{k=1}^{n}(1-P(e_k))]$$

$$\overline{P}(0) = 1 - \sum_{i=1}^{n}\overline{P}(\sigma_i) - \sum_{j=1}^{n}\overline{P}(e_j)$$

4- Compute $P_{ij}(a) = \sum_{\eta\in\Omega}\overline{P}(\eta)$ where Ω is the set of all possible events (sensory or plant related) which are possible at state $Y'(t)$. Let $R(Y(t),a)$ be the reward earned by making decision a, $Y(t) \rightarrow Y'(t)$, when the current state is $Y(t)$. We define $R(Y(t),a) = \begin{cases} 1 & \mathit{If\ Y'(t)\ sets\ the\ system} \\ & \mathit{to\ model.} \\ 0 & \mathit{Otherwise} \end{cases}$ Using MDP theory [Ross 97], the optimal switching policy for the stated system is determined from the solution of the following linear programming model:

$$\begin{array}{lll} Max & \sum_i \sum_a R(i,a)\pi_{ia} & \\ S.T. & \pi_{ia} \geq 0 & \forall i,a \\ & \sum_i \sum_a \pi_{ia} = 1 & \\ & \sum_a \pi_{ja} = \sum_i \sum_a \pi_{ia} P_{ij}(a) & \forall j \end{array}$$

where π_{ia} denote the steady state probability of being in state i and selecting action a. It can be proven that the optimal policy is a steady state one and is determined by: $\beta_i^*(a) = \frac{\pi_{ia}^*}{\sum_a \pi_{ia}^*}$ where $\beta_i^*(a)$ is the probability of making decision a in state i and π_{ia}^*'s are optimal solutions of the LP model. It is obvious that, due to the steady state policy, $\forall\ 1 \leq i \leq n\ \exists\ a^* : \beta_i^*(a^*) = 1\ and \sum_a \beta_i^*(a) = 1$.

We illustrate the above model by the following example.

4-Example

Let $K = L(S/G)$ be the coupled language of the supervisor graph of Figure 2. Here, we do not need to describe the plant language or feedback policy. Let controllers of Figure 3 and Figure 4 be two reduced supervisors, constructed using projections P_2 and P_3, where $\Sigma_o^{P_1} = \{a,b,c\} = \Sigma$, $\Sigma_o^{P_2} = \{a,b\}$ and $\Sigma_o^{P_3} = \{a,c\}$. We assume $O = \{P_1, P_2, P_3\}$. The set of states and the possible actions in each state are given in Table 1. Also possible events in each state are given in Table 2. It is noted that since event a must always be observed, its sensory device state is not included in state expression. We let $\lambda_{a/1} = \lambda_{a/4} = \lambda_{a/6} = 10$, $\lambda_{b/2} = \lambda_{b/5} = 12$, $\lambda_{c/3} = \lambda_{c/7} = 11$, $f_b = 5$, $f_c = 2$, $r_b = 7$, $r_c = 6$, $B_{1/4} = 0.8$, $B_{3/4} = 0.2$, $B_{2/7} = 0.3$ and $B_{3/7} = 0.7$. Table 3 shows the optimal switching policy obtained by MDP for the system of this example.

5-Conclusion

MDP modeling of controller switching of discrete event systems was introduced. The optimal solution of the model determines the switching policy, which maximizes the percentage of time that the coupled system of the plant and controller act in its model, i.e. it generates a normal language.

6-References

[Cieslak et al. 88] Cieslak, R., Desclaux, C., Fawaz, A.S., and Varaiya, P., "*Supervisory control of discrete-event processes with partial observations*", IEEE Transactions on Automatic Control, vol. 33, no. 3, pp. 249-260, 1988.

[Darabi and Jafari 98] Darabi, H. and Jafari M. A., " "*Supervisory Control and Optimal Partial Observation Problem*", Report 98-142, Dept. of Industrial and Systems Engineering, Rutgers University, 1998.

[Haji-Valizadeh and Loparo 96] Haji-Valizadeh A. and Loparo K. A., *"Minimizing the Cardinality of an Events Set for Supervisors of Discrete-Event Dynamical Systems"*, IEEE Transactions on Automatic Control, vol. 41, no. 11, pp. 1579-1593, 1996.

[Lin and Wonham 88] Lin, F. and Wonham, W.M., *"Decentralized supervisory control of discrete-event systems"*, Information Science vol. 44, no. 3, pp.199-224, 1988.

[Ramadge and Wonham 87] Ramadge, P.J. and Wonham, W.M., *"Supervisory control of a class of discrete event Processes,"* SIAM J. Contr. Optimiz., vol. 25, no.1,pp. 206-230, 1987.

[Ross 97] Ross S. M., Introduction to Probability Models, Sixth edition, Academic Press, San Diego CA, 1997.

[Vaz and Wonham] Vaz A. F. and Wonham W. M., *"On Supervisor Reduction in Discrete-event Systems"*, Int. J. of Control, vol. 44, no. 2, pp.475-491, 1986.

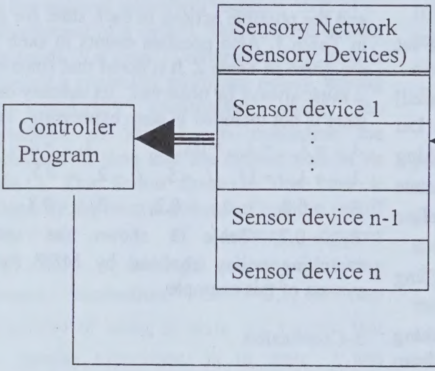

Sensory Network
(Sensory Devices)
Sensor device 1
:
:
Sensor device n-1
Sensor device n
Controller Program
Plant Generator

Figure1. Control Loop

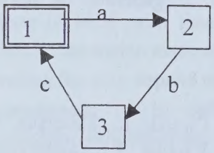


Figure 2. Supervisor for Projection P_1

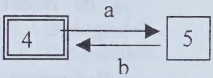


Figure 3. Supervisor for Projection P_2

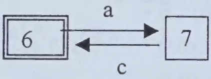


Figure 4. Supervisor for Projection P_3

State*	State No.	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
II0	**1**	**0**				**1**			**2**				**3**			
IR0	**2**		**0**								**4**					
RI0	**3**			**0**											**5**	
RR0	**4**				**0**											
WW1	**5**	**6**				**0**			**7**				**8**			
WW2	**6**	**9**					**0**			**10**				**11**		
WW3	**7**	**12**						**0**	**13**					**14**		
WI4	**8**	**15**							**0**							
WI5	**9**	**16**					**17**			**0**				**18**		
WR4	**10**		**19**								**0**					
WR5	**11**		**20**									**0**				
IW6	**12**	**21**				**22**			**23**				**0**			
IW7	**13**	**24**												**0**		
RW6	**14**			**25**											**0**	
RW7	**15**			**26**												**0**

Table 1: MDP States and Switching Decisions

First and second columns show the MDP states descriptions and index numbers respectively. First row is the same as second column. Other numbers in the table are decision numbers. For example decision number 6 (third column- sixth row) shows a switching action from MDP state 5 to MDP state 1.

	a	b	c	b failure	c failure	b repair	c repair
1							
2							1
3						1	
4						1	1
5	1			1	1		
6		1		1	1		
7			1	1	1		
8	1			1			
9		1		1			
10	1			1			1
11		1		1			1
12	1				1		
13			1		1		
14	1				1	1	
15			1		1	1	

Table 2: Possible Events in each State

State	Optimal Action
1	2
2	4
3	5
4	0
5	7
6	10
7	13
8	0
9	0
10	0
11	0
12	23
13	0
14	0
15	0

Table 3: Optimal Policy (Ex.: In MDP state 5 the best action is to switch to MDP state 8. This action is numbered as 7 in Table 1)

Supervisory Control of Contradictions in Hierarchical Task Controllers

Xiaoyi Guan
Dept. of Electrical Engineering
University of Kentucky
Lexington, KY 40506
xguan01@engr.uky.edu

Lawrence E. Holloway
Center for Manufacturing Systems and
Dept. of Electrical Engineering
University of Kentucky
Lexington, KY 40506-0108
holloway@engr.uky.edu

Abstract In this paper we consider the supervisory control of automatically generated control logic. Specifically, we consider generated control logic for sequencing activities in discrete event systems. The logic consists of modules called taskblocks, which interact in a hierarchical manner but may also operate concurrently. The purpose of supervisory control is to prevent these concurrent taskblocks from outputting contradictory control signals to the plant. In this paper, we present a method for automatically synthesizing the supervisory control system to regulate the activities in the generated control logic.

I. INTRODUCTION

The supervisory control framework for discrete-event systems developed by Ramage and Wonham [1] separates the supervisor from the process. Given a process, a supervisor is designed such that the supervised process couples with the supervisor and behaves according to the given specification. In contrast, Charbonnier, Alla, and David present a *supervised control* concept to separate control and supervision[2]. With their approach, the controller is responsible for sequencing and forcing some events to occur in the plant, and the supervisor then operates on the controller to prevent it from generating some events.

In this paper, we consider a supervised control structure similar in concept to that of Charbonnier, Alla, and David. In particular, we consider a framework as shown in Figure 1, where a controller works directly on the plant, and a supervisor then controls the controller.

The controller that we consider is a discrete event system that interacts with the plant through binary truth signals called*conditions.* The controller is composed of many pieces of control logic, called *taskblocks.* These taskblocks are generated in modular manner for a given specification using a model of the system. They interact with each other and with the plant through condition signals. The controller receives state information from the plant through the set of conditions C_{state}, and the set of conditions $C_{interface}$ are used for the controller to communicate actuation information to the plant.

Many taskblocks in the controller may be working together hierarchically, sequentially, or concurrently. Because of this concurrency and the manner in which the taskblocks are created automatically, it is possible that the control system will try to do contradictory actions in the plant. For example, one taskblock may be trying to turn off a motor at the same time that another taskblock is working to keep it on. The purpose of the supervisor is to regulate the activities of the controller to prevent it from outputting contradictory signals to the plant. The supervisor interacts with the controller through input conditions $C_{activation}$ and output conditions C_{super} without direct connection to plant.

In the next section, we overview the structure of the system and the condition system modeling framework. In section III, we discuss taskblocks and the structure of the controller. Section IV presents a supervisor to prevent contradictions from the controller. Section V presents a summary of the work.

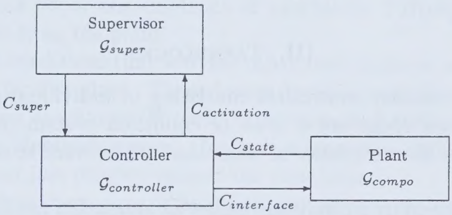


Fig. 1. Supervisory Control of Task Controller

II. STRUCTURE OF CONTROL SYSTEM

In this paper, we consider systems modeled as *condition systems* [3; 4; 5]. A *condition* is a binary signal that has value either “true” or “false”. A *condition system* is defined as a form of Petri net that interacts with other systems by these condition signals. The condition systems we consider are a class of the condition-event models introduced by Sreenivas and Krogh [9]. Rausch, Luder, and Hanisch consider control synthesis for forbidden state avoidance for condition systems[6; 7].

A Petri net G defining the dynamics of the condition system consists of the usual places, transitions, and arcs, but is augmented by a condition mapping function, $C_G()$, over its places and transitions. The conditions associated with places allow the system to output information to outside systems: if a place p is marked under a given state, then all conditions $c \in C_G(p)$ are output true. The conditions associated with transitions allow conditions from outside the system to influence the system behavior: given a transition t, the transition can only fire if all of its conditions $c \in C_G(t)$ are true. Thus, the conditions associated with places determine the output of the system based on its state, and the conditions associated with transitions represent inputs that influence the state evolution. Given a condition system G, its output condition set is defined as $C_{out}(G) = \{c \in C_G(p) \mid p \in G\}$, and input condition set is defined as $C_{in}(G) = \{c \in C_G(t) \mid \acute{\imath} \in G\}$.

Conditions can be combined together in logic expressions using the operators ”AND” and ”OR”. $\vee$ and $\wedge$ are used to denote logic OR and logic AND in this work, respectively.

As shown in figure 1, the systems we consider are represented by collections of condition models. These models are partitioned into three sets: the set of *component models*, denoted $\mathcal{G}_{compo}$, the set of *taskblock models*, denoted $\mathcal{G}_{controller}$, and the set of *supervisory models*, denoted $\mathcal{G}_{super}$. The component models represent the plant, the taskblock models represent the controller logic, and supervisory models represent the control supervisor. In this paper, we assume that the taskblock models are all synthesized automatically and thus have specific forms, as detailed in [8]. Each taskblock model represents specific portions of control logic for sequencing the plant through desired states. The taskblock models are modular, they run concurrently, and they activate each other in a hierarchical manner. The supervisor models, which we focus on in this paper, supervise the control logic, ensuring that the concurrent control activities do not result in contradictory control signals to the plant.

The set of conditions output from $\mathcal{G}_{super}$ and input to $\mathcal{G}_{controller}$ is denoted C_{super}. These are signals in which the supervisor enables or disables the activation of taskblocks in the controller. The controller then indicates to the supervisor which taskblocks are active. This is done through the set of conditions $C_{activation}$. $\mathcal{G}_{controller}$ drives actuators in the plant $\mathcal{G}_{compo}$ with the set of signals $C_{interface}$, and observes the system through the set of conditions C_{state}.

III. TASKBLOCKS

In this paper, we consider controllers consisting of individual pieces of control logic called *taskblocks.* Taskblocks are a type of condition system defined in [5]. In this paper, we only discuss characteristics of them that are relevant to our supervisory control problem.

A taskblock is identified by a condition, called an *activation condition*, which is an input only to it. Suppose do_v is an activation condition, then we denote the taskblock associated with that condition as $TB(do_v)$. Let C_{do} indicate the set of activation conditions for all taskblocks. Each taskblock includes a designated *idle place* which outputs condition $idle(do_v)$. A taskblock with the idle place marked is considered idle, and one without the idle place marked is considered *activated.* The output transition of the idle place is called the *activating transition*, and has as inputs the condition do_v and the condition $sup(do_v)$, which is a condition input from the supervisor. Thus, an idle taskblock can only become activated if the conditions do_v and $sup(do_v)$ are both true. The taskblock has a structure such that whenever do_v becomes false, the taskblock returns to idle.

Figure 2 shows an example taskblock. This is a taskblock with name $TB(do_v)$ associated with the activation condition $do_v \in C_{do}$. p_1 is idle place. The transition t_1 is the activating transition. If the taskblock is idle and the conditions on t_1 are true, then t_1 fires and $TB(do_v)$ is activated. The Petri net structure of the taskblock defines the control logic. When the plant or other taskblocks output conditions that enable transitions to fire (such as t_2 in the figure), then the marking of the taskblock changes, and the taskblock then outputs conditions according to the new marking. These outputs then either activate other taskblocks or are inputs to the plant.

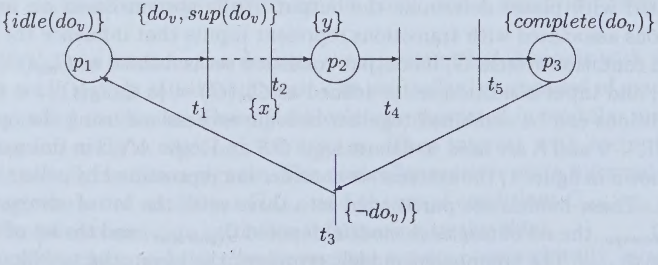


Fig. 2. An example taskblock structure.

For any condition $x \in C_{out}(G_{compo})$ output from the plant, we assume the existence of two types of taskblocks associated with x.

- An *Action-type* taskblock for x has activation condition denoted do_x^A, and is denoted by $TB(do_x^A)$. Its purpose is to drive the plant to a given condition, x, from any initial state.
- A *Maintain-type* taskblock for x has activation signal denoted do_x^M, and is denoted $TB(do_x^M)$. Its purpose is to keep the condition x of the closed-loop system true, given that x was already true when the taskblock was activated.

The references [8; 5] give criteria for the existence of action and maintain taskblocks, and algorithms to generate them from analysis of component models. Based on this prior work, we assume in this paper the existence of taskblocks $TB(do_x^A)$ and $TB(do_x^M)$ for any condition x output from the plant.

A taskblock outputs conditions that activate other taskblocks or are directly translated into actuation signals for the plant. These lowest level signals that feed directly to the plant are called *interface conditions.* For such conditions do_x^A and do_x^M and the condition x itself are effectively identical signals. In such a case the taskblocks $TB(do_x^A)$ and $TB(do_x^M)$ are trivial and just directly output the condition x.

According to task functions, a taskblock always outputs an action-type activation condition to make the condition true before it outputs maintain-type activation conditions to keep this condition true. Therefore, an action-type taskblock must be completed to output a condition before a sequence of maintain-type taskblocks are activated to maintain the condition. For each maintain-type taskblock activation condition do_v^M, let $\mathit{Action}(do_v^M) := do_v^A$ denote the action-type activation condition signal for the condition v.

The task controller, $\mathcal{G}_{controller}$, is assembled from a specification net, $\mathcal{G}_{spec}$ and taskblocks, where $\mathcal{G}_{spec}$ is a condition system that defines desired system behaviors by outputing sequences of (high-level) taskblock activation conditions. $\mathcal{G}_{controller}$ includes all possible activated taskblocks in order to complete the given specification $\mathcal{G}_{spec}$. The activation procedure begins with the output conditions of $\mathcal{G}_{spec}$. When an output condition is true, its associated taskblock will be activated if it is in idle state and its activating transitions are enabled. In turn, activated taskblocks output conditions which activate other associated taskblocks until finally interface conditions are output. This activation of taskblocks is in a hierarchy structure from the top conditions in specification net to interface conditions. An acyclic directed graph can be used to represent this activation of taskblocks as shown in figure 3. The graph is called *hierarchical graph.* A hierarchical graph begins from top level taskblocks and ends with interface conditions. Note that the graph does not indicate any sequencing information among outputs of a taskblock, it only indicates that a given taskblock may at some time output certain conditions.

In figure 3, rectangles represent taskblocks and polygons denote output conditions of taskblocks. N_0 and N_1 are specification nets, $\mathcal{G}_{spec} = \{N_0, N_1\}$, N_0 has three activation conditions, do_a, do_b, and do_c. N_1 outputs conditions do_b and do_d. N_0 and N_1 have one common activation condition do_b. Conditions e, g, h, j, and k are interface conditions. $\mathcal{G}_{controller} = \{\, N_0, N_1, TB(do_a), TB(do_b), TB(do_c), TB(do_d), TB(do_f), TB(do_m)\}$. There are several paths through the graph from the specification nets to interface conditions. These paths are called *activation paths.* Examples of activation paths include $N_0 - do_a - TB(do_a) - e$ and $N_0 - do_a - TB(do_a) - do_f - TB(do_f) - j$. Define $\Pi(\mathcal{G}_{controller})$ as the collection of activation paths of the controller $\mathcal{G}_{controller}$. During a control process, in general, only some of taskblocks of the controller are activated at any given state. Recall

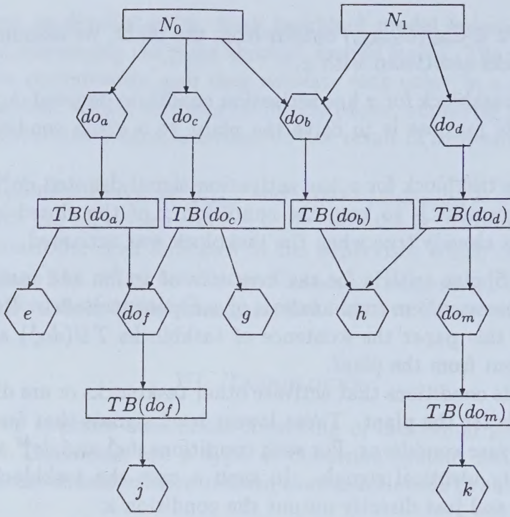


Fig. 3. A hierarchy graph.

that a taskblock is activated only while its activation condition is true. The activation path indicates a sequence of taskblocks which can output activation conditions to lower taskblocks in the sequence. Thus, a taskblock in the graph is activated only if there is some activation path which contains the taskblock and for which all higher taskblocks in the path are also activated.

When enforcing a given specification on a controlled plant, the controller may have contradictory conditions to be true at some marking state because of its concurrent operation. If these conditions are interface condition, then this is not acceptable, since it is impossible that contradictory commands can be simultaneously executed in the plant. Therefore, supervision is required to monitor the controller to prevent it from outputting any pair of contradition interface conditions at any marking state. That means task controllers need to be controlled by a supervisor, as shown in figure 1.

As indicated in the previous section, each taskblock $TB(do_x)$ has a supervisory control input condition $sup(do_x)$ that is able to prevent it from becoming activated. However, we now impose restrictions on when such supervisory action is permitted. First, note that action-type taskblocks can steer the system to the target condition under any initial system state, so any delay that the supervisor imposes on the action-type taskblock would not affect its ability to reach the target. Maintain-type taskblocks, however, represent control logic to keep a target condition true, but are defined such that it requires the condition to already be true when the taskblock is activated. Thus, if the supervisor imposes any delay in activating the maintain-type taskblock or any taskblock indirectly activated by the maintain-type taskblock, then the plant may have already moved beyond the target condition and the control logic may not be valid.

Our supervision requirements can now be summarized as follows:

- Supervision Requirement SR1: The supervisor should prevent the controller from outputing any contradictory interface conditions.

- Supervision Requirement SR2: The supervisor should not prevent activation of any control activity associated with maintaining a condition. This implies it should not directly prevent a maintain-type taskblock for which its activation condition is true, and should not prevent activation of any taskblock directly or indirectly activated by the maintain-type taskblock.

The above requirements immediately lead to the result that for any maintain-type taskblock $TB(do_x^M)$ with activation condition do_x^M, the supervisor must not prevent activation of the taskblock, so $sup(do_x^M)$ is true always. Furthermore, the above assumption implies that if an action-type taskblock is being called directly or indirectly by a maintain-type taskblock, the action-type taskblock should not be suspended by the supervisor. Thus, given a path in the hierarchy graph from the specification net to an interface condition, if there is a maintain-type taskblock in the path, then we must not disable any action blocks below it in the path. This leads us to the following definition:

Definition 1 : Uncontrollable Boundary Point and Control Block
Consider a hierarchy graph and a path π in the graph such that the path ends with some interface condition x.

- *The* uncontrollable boundary condition *for π is the earliest point in π such that if the condition is true, then under supervision requirement SR2, the supervisor cannot prevent the output of condition x.*
- *The* control block *for π is a taskblock that (1.) can be kept idle under the supervisor, and (2.) which if it is not kept idle then there are no other controls (under SR2) that can prevent x from becoming true.*

Note in the above definition, the control block is not required to be a taskblock in the path π. In fact, as shown in the following lemma, it will often be an action-type taskblock associated with the highest maintain-type taskblock in the path, and so is often not in the path itself.

Lemma 1 : *Consider an activation path π ending with an interface condition x.*

1. *If π does not contain any maintain-type activation conditions, then the uncontrollable boundary point of π is x itself. The control block of π is the action-type taskblock which outputs x (the last taskblock in π).*
2. *If π contains maintain-type conditions, then the uncontrollable boundary point of π is the first maintain-type activation condition in π. Let this be denoted do_z^M. Then the control block for π is the action-type taskblock $TB(do_z^A)$.*

In the lemma 1, we may notice that the action-type activation condition $do_z^A = Action(do_z^M)$. According to the lemma 1, the control block for x through π in figure 4(a) is $TB(do_w^A)$ and the control block for x through π' in figure 4(b) is $TB(do_v^A)$ corresponding to the uncontrollable boundary point do_v^M.

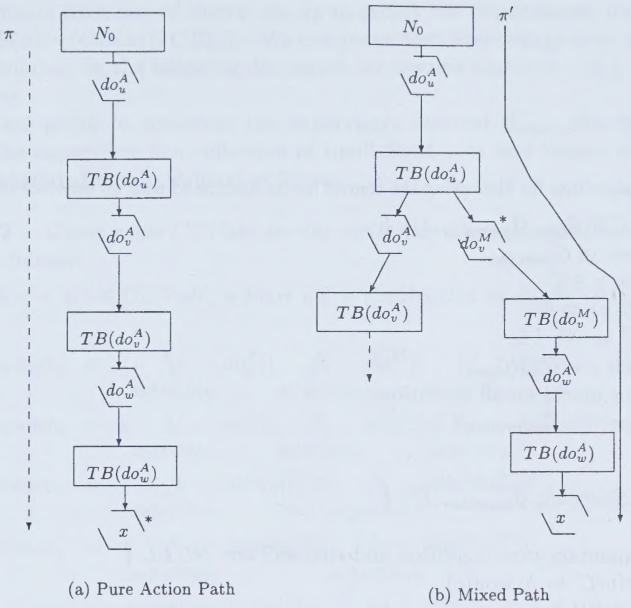
Proof: For the controller $\mathcal{G}_{controller}$, given an activation path $\pi \in \Pi(\mathcal{G}_{controller})$. Let $\pi = TB(do_{c_0})do_{c_1}TB(do_{c_1})do_{c_2}TB(do_{c_2})\ldots do_{c_{n-1}}TB(do_{c_{n-1}})do_{c_n}TB(do_{c_n})x$, where $do_{c_i} \in C_{do}$ represents activation conditions, $TB(do_{c_i}) \in \mathcal{G}_{controller}$ represents the taskblocks associated with by do_{c_i} for $i = 1, 2 \ldots, n$, and x is an interface condition. $TB(do_{c_n})$ outputs x. We prove the lemma 1 in two cases as follows:

1. Case 1: Every activation condition $do_{c_i} \in \pi$ is action-type
   We know that the activation of the action-type taskblock $TB(do_{c_n})$ can be controlled with the input conditions of its activating transitions, do_{c_n} and $sup(c_n)$. If do_{c_n} is true, setting $sup(c_n)$ to be false can keep $TB(do_{c_n})$ in the idle state. Therefore, the interface condition x will not be output by π. Thus, $TB(do_{c_n})$ is the control block for π. On the other hand, if x is true, this implies that the place outputing x is marked. Therefore, in this case, x is the uncontrollable boundary point for π.
2. Case 2: π contains maintain-type activation conditions
   Let do_{c_i} is an actiion-type condition for $i = 0 \dots k-1$ and do_{c_k} is a maintain-type condition, $1 \leq k \leq n$. do_{c_k} is the first maintain-type condition in π. If do_{c_k} is true, the maintain-type taskblock $TB(do_{c_k})$ must be activated since the controller requires the system to stay at the current state. That means any maintain-type task is not allowed to block, but its corresponding action-type task can be stopped by disabling activating transitions. Any taskblock lower than $TB(do_{c_k})$ in π must be activated in order to complete $TB(do_{c_k})$ under requirement SR2. Therefore, do_{c_k} is the uncontrollable boundary point of x through π. Since any sequence of maintain-type taskblocks always follows its corresponding action-type taskblock, $TB(Action(do_{c_k}))$ must be activated before $TB(do_{c_k})$ to output the condition c_k. We may disable the activating transition of the action-type taskblock $TB(Action(do_{c_k}))$ to prevent the maintain-type taskblock $TB(do_{c_k})$ from being activated, in turn, to prevent x from being output through π. So we prove that $TB(Action(do_{c_k}))$ is the control block of x through π for this case.□□□

◇◇◇

The definition 1 divides a activation path into two regions, one is controllable and the other is uncontrollable. For any activation path ending with the interface condition x, there are two possibilities: π does not have any maintain-type conditions or π includes some maintain-type conditions. For the first case shown in figure 4(a), do_u^A, do_v^A, and do_w^A are all action-type conditions and x is an interface condition. The activation path π has the uncontrollable boundary point x for the condition x, and has control block of $TB(do_w^A)$. For the second case shown in figure 4(b), π' has a maintain-type condition do_v^M. do_v^M is the uncontrollable boundary point for the interface condition x in π', and the control block is $TB(do_v^A)$.

Given a controller, an interface condition may have many uncontrollable boundary points and control blocks, since the interface condition may appear in many activation paths. An algorithm is developed to search a controller to find all control blocks which may be activated in the control for an interface condition. Given the controller $\mathcal{G}_{controller}$, specification net $\mathcal{G}_{spec}$, and a set of interface conditions C, we want to find control blocks for any condition $c \in C$. In the following algorithm, *ActiveC* denotes the activation condition associated with the control block for the current searching path. $CB(x)$ denotes the set of activation conditions associated with the control blocks in $\mathcal{G}_{controllor}$ for the interface condition $x \in C$. The flag *UBP* denotes the uncontrollable boundary point. The algorithm is based on the depth first search.



(a) Pure Action Path

(b) Mixed Path

* indicates uncontrollable point

Fig. 4. Uncontrollable Boundary Point On Activation Path

With the algorithm, we can find all control blocks for any interface condition in a controller. Then, supervisory conditions can be output to these control blocks to control the interface condition. In the next section, we will discuss the supervisory control on the contradiction interface conditions of controllers.

IV. Supervisory Control of Contradiction State of Plant

In the previous section, we developed an algorithm to determine control blocks for interface conditions of a controller. In this section, we will develop a supervisory control for contradiction state avoidance. With this supervisory control, no contradiction conditions are output to the plant. Given a controller $\mathcal{G}_{controller}$, the set of contradiction condition pairs, denoted $CCPair$, is defined as $CCPair = \{< x, y >| \; x, y \in C_{interface}$, x and y are contradiction conditions $\}$. Let $CCSet = \{u \mid u \in < x, y >$ for $< x, y > \in CCPair\}$, where $CCSet \subseteq C_{interface}$. Before we propose our supervisory control, we need an assumption on control blocks for any pair of contradiction conditions $< x, y > \in CCPair$.

Assumption 1 Conflict Free
Given a pair of contradiction interface conditions $< x, y > \in CCPair$, $CB(x)$ *and* $CB(y)$ *are said to be* conflict free *if* $CB(x) \bigcap CB(y) = \emptyset$. *We assume all pairs in* $CCpair$ *are conflict free.*

The assumption 1 requires that any pair of contradiction conditions can not be controlled by a single taskblock. If $CB(x) \bigcap CB(y) \neq \emptyset$ for $< x, y >$, we can remove the

Fig. 5. An algorithm for identifying the control blocks associated with an interface condition.

```
Procedure: $CB(\mathcal{G}_{spec}, \mathcal{G}_{controller}, C)$ {
// $C$ is a subset of $C_{interface}$
    For each $x \in C$
        $CB(x) \Leftarrow \emptyset$;
    $ActiveC \Leftarrow NULL$;
    For each $c \in C_{out}(\mathcal{G}_{spec})$
        clear marks on all conditions;
        $Search(c, \mathcal{G}_{controller}, C)$;
}

Procedure: $Search(c, \mathcal{G}_{controller}, C)$ {
    mark c;
    if c is maintain-type condition and $ActiveC == NULL$ {
        $ActiveC \Leftarrow$ Action(c);
        Set UBP flag on $c$;
    }
    For each condition $c' \in C_{out}(TB(c))$ {
            Case: {
                If $c'$ is unmarked
                    $Search(c', \mathcal{G}_{controller}, C)$;
                If $c' \in C$ {
                    If( $ActiveC == NULL$)
                        $CB(c') \Leftarrow CB(c') \cup \{c\}$;
                    else
                        $CB(c') = CB(c') \cup \{ActiveC\}$;
                }
                If $c'$ has UBP flag set {
                    $ActiveC \Leftarrow NULL$;
                    clear UBP flag on $c'$;
                }
            }
    }
}
```

common elements from one of control blocks to satisfy the requirement, for instance let $CB(x) = CB(x) - (CB(x) \bigcap CB(y))$. We can prove that this change does not affect the supervisory control. In the following discussion, we assume that any $< x, y > \in CCPair$ is conflict free.

Now, we are going to construct the supervisor, denoted $\mathcal{G}_{super}$, for the controller $\mathcal{G}_{controller}$. The supervisor is a collection of small Petri nets and binary expressions of conditions, which is formally defined as follows.

Definition 2 *: Given some $CCPair$ for the controller $\mathcal{G}_{controller}$, the supervisor $\mathcal{G}_{super}$ is defined as follows:*

1. *For each $< x, y > \in CCPair$, a Petri net is constructed as shown in figure 6, where*

$$request_x = \Big(\bigvee_{do_u^A \in CB(x)} do_u^A\Big)\Big(\bigwedge_{do_v^A \in CB(y)} \overline{do_v^M}\Big)\Big(\bigvee_{do_u^A \in CB(x)} do_u^M\Big) \tag{1}$$

$$request_y = \Big(\bigvee_{do_v^A \in CB(y)} do_v^A\Big)\Big(\bigwedge_{do_u^A \in CB(x)} \overline{do_u^M}\Big)\Big(\bigvee_{do_v^A \in CB(y)} do_v^M\Big) \tag{2}$$

$$release_x = \Big(\bigwedge_{do_u^A \in CB(x)} idle(do_u^A)\Big)\Big(\bigwedge_{do_{u'}^A \in CB(x)} idle(do_{u'}^M)\Big) \tag{3}$$

$$release_y = \Big(\bigwedge_{do_v^A \in CB(y)} idle(do_v^A)\Big)\Big(\bigwedge_{do_{v'}^A \in CB(y)} idle(do_{v'}^M)\Big) \tag{4}$$

2. *For any action-type taskblock $do_u \in C_{Action}$, let*

$$ControlledCondition(do_u) = \{z \in CCSet \mid do_u \in CB(x)\}$$

If $ControlledCondition(do_u) \neq \emptyset$, let

$$sup(do_u^A) = \bigwedge_{z \in ControlledCondition(do_u)} go_z$$

If $ControlledCondition(do_u) = \emptyset$,

$$sup(do_u^A) = 1$$

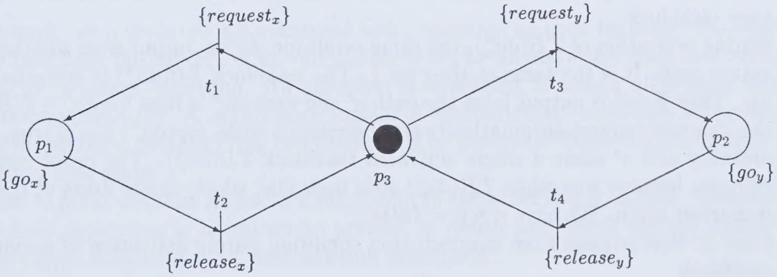


Fig. 6. supervision net for contradiction conditions $\{x, y\}$

In the definition 2, we may notice that any pair of contradiction conditions share a single token in $\mathcal{G}_{super}$, which means that two contradiction conditions can not become true at any marking state simultaneously. $\mathcal{G}_{super}$ also outputs conditions to each action taskblock to control if a taskblock can be activated or not. We want to point out that if

$ControlledCondition(do_u) = \emptyset$, then all interface conditions controlled by $TB(do_u)$ are not contradiction conditions. Therefore, $TB(do_u)$ can always be condition enabled by $sup(u) = 1$ if do_u is true. The interaction between $\mathcal{G}_{controller}$ and $\mathcal{G}_{super}$ through conditions makes the plant run without contradiction. The following theorem gives the main result about the supervisory control of contradiction conditions.

Theorem 1 *: Consider a control system $\mathcal{G} = \mathcal{G}_{compo} \bigcup \mathcal{G}_{controller} \bigcup \mathcal{G}_{super}$ with contradiction conditions $CCPair$ such that all pairs are conflict free. For any $m \in M_{\mathcal{G}}$, the subset of $C_{interface}$ output under the marking m is contradiction free.*

The theorem 1 means that $\mathcal{G}_{super}$ will regulate the behavior of $\mathcal{G}_{controller}$ such that no contradiction conditions are sent to the plant $\mathcal{G}_{plant}$. We prove this theorem in following.

Proof

Since $C_{interface} \subset C_{out}(\mathcal{G}_{controller})$, we can restate the theorem as follows: for any pair $< x, y > \in CCPair$, the supervisor $\mathcal{G}_{super}$ can keep $x \wedge y = false$ at any marking state in $\mathcal{G}_{controller}$. We prove this statement.

Given $\mathcal{G}_{controller}$ with $CCPair$, for $\forall < x, y > \in CCPair$, we want to prove $x \wedge y = false$ at any marking state. For x and y, $CB(x)$ and $CB(y)$ can be obtain by using the algorithm. We will show $x \wedge y = 0$ in any possible situation in order to prove the theorem. The figure 6 is used to assist this proving.

1. Case 1: Single request of action-type activation condition
   Consider a marking m_0 such that do_u^A is output from a single path π for $do_u^A \in CB(x)$ and each do_v^A is false for $do_v^A \in CB(y)$. For $\mathcal{G}_{super}$, the evalution of input conditions makes $request_x = true$ and $request_y = false$, which enables t_1 of figure 6. The firing of t_1 leads to the marking state m_2 such that $m_2(p_1) = 1$ and condition go_x becomes true. For $x \in ControlledCondition(do_u^A)$, the evaluation of expression $sup(do_u^A)$ results in $sup(do_u^A) = 1$, which enables the activation transition of the taskblock $TB(do_u^A)$ since do_u^A and $sup(do_u^A)$ both are true. Since each do_v^A is false for $TB(do_v^A) \in CB(y)$, then there is no contradiction condition to appear against x. We have $x \wedge y = 0$.
2. Case 2: New request for the same condition during activation of associated action-type taskblock
   During activation of $TB(do_u^A)$, the same condition do_u^A is output from another activation path. It is the same as the case 1. The taskblock $TB(do_u^A)$ is activated after m_2. Then if do_u^A is output from the path π' and each do_v^A is false for $do_v^A \in CB(y)$ at m_3, this new request automatically gets permission since $sup(do_u^A)$ is still true, which means π and π' share a single activated taskblock $TB(do_u^A)$. The input condition $release_x$ become true when $TB(do_u^A)$ goes back idle, which causes firing of t_2 and p_3 is marked again. We have $x \wedge y = false$.
3. Case 3: New request from contradiction condition during activation of action-type taskblock
   Condition do_v^A for $do_v^A \in CB(y)$ is while $TB(do_u^A)$ is activated. Suppose a system marking such that do_u^A is output from a single path π for $do_u^A \in CB(x)$ and each do_v^A is false for $do_v^A \in CB(y)$, then from case 1, $m(p_1) = 1$ and condition go_x is true. Thus, $x \in ControlledCondition(do_u^A)$ and $sup(do_u^A) = 1$ and the taskblock $TB(do_u^A)$ is activated. Now suppose do_v^A becomes true for $do_v^A \in CB(y)$. The evaluation of input conditions of $\mathcal{G}_{sup}$ make $request_y = 1$. The transition t_3 is still disabled since

the place p_3 is unmarked, which results in $go_y = false$. Thus, $sup(do_v^A)$ is evaluated to be false. The taskblock $TB(do_v^A)$ can not be activated until the place p_3 gets the token. On the other hand, p_3 cannot be marked until $TB(do_u^A)$ and $TB(do_u^M)$ go back to the idle state which means that the condition x is false. Then, $x \wedge y = 0$.

4. Case 4: Request from maintain-type conditions after activation of action-type taskblock
   Let maintain-type condition do_u^M follow the action-type condition do_u^A. Consider again a marking following case 1 where do_u^A is output from a single path π for $do_u^A \in CB(x)$ and each do_v^A is false for $do_v^A \in CB(y)$. From case 1, $m(p_1) = 1$ and condition go_x is true. For $x \in ControlledCondition(do_u^A)$, the evaluation of expression $sup(u)$ results in $sup(do_u^A) = 1$, which enables the activation transition of the taskblock $TB(do_u^A)$ since do_u^A and $sup(do_u^A)$ both are true, thus activating taskblock $TB(do_u^A)$. Now suppose that the system marking changes so now do_v^A is true for $do_v^A \in CB(y)$. The evaluation of input conditions of $\mathcal{G}_{sup}$ make $request_y = 1$. But the transition t_3 is still disabled since the place p_3 is unmarked, which results in $go_y = false$ so $sup(do_v^A)$ is false and $TB(do_v^A)$ remains in an idle state. Assume that $TB(do_u^A)$ completes and returns back to idle state $idle(do_u^A)$ at m_k. The evaluation of the input conditions of $\mathcal{G}_{super}$ has $request_x = true$ since do_u^M is output, and $request_y = false$ since $\overline{do_u^M}$ blocks it. Therefore, $TB(do_u^M)$ is activated and $TB(do_v^A)$ is kept in idle state until $TB(do_u^M)$ completes. We have $x \wedge y = 0$.
5. Case 5: Competition from contradiction conditions
   Let $do_u^A \in ControlBlock(x)$ and $do_v^A \in CB(y)$. Given a marking state m_0 such that do_u^A is output for $do_u^A \in CB(x)$ and do_v^A is output for $do_v^A \in CB(y)$, the evaluation of the input conditions of $\mathcal{G}_{super}$ has $request_x = true$ and $request_y = true$. Transitions t_1 and t_3 are both enabled. But since only one token is available, the result of competition is only one oftransition fires. go_x or go_y will output, which makes $TB(do_u^A)$ or $TB(do_v^A)$ be activated. Therefore, we have $x \wedge y = 0$.

According to the analysis above, the supervisory control $\mathcal{G}_{super}$ always can make $x \wedge y = 0$, which implys $g_{\mathcal{G}}(m) \bigcap C_{interface}$ is contradiction free for any marking m.

◇◇◇

V. Conclusion

In this work, we introduced a hierarchical task controller, analyze its controlstructure. Then, we introduce the concept of uncontrollable point for an activation path and control blocks for an interface condition. An algorthm is developed to search all control block for an interface condition. A supervisory control of contradiction conditions is proposed. Any pair of interface contradicting conditions can be prevented from outputing true at the same marking state.

We want to point out that a deadlock may occur in the supervisory control. Therefore, a higher level supervisor is necessory to prevent or detect and break up any deadlock caused by supervisory control of contradiction conditions.

REFERENCES

[1] P. J. Ramadge and W. M. Wonham, "Supervisory control of a class of discrete-event processes", *SIAM J. on Control an Optimization*, Vol. 25, pp.206-230 Jan, 1987.

[2] F. Charbonnier, H. Alla, and R. David, "Supervised control of discrete-event dynamic systems", *IEEE Trans. on Control Systems Technology*, Vol. 7, No. 2, pp.175-187, March, 1999.

[3] L. E. Holloway and J. Ashley Jr. *Condition languages and condition systems for modeling ambiguous control specifications.* In IEE international Workshop on Discrete Event Systems (WODES98), Cagliari, August 1998.

[4] L. E. Holloway and J. Ashley Jr. *Elaborative orderings of condition languages.* In preceedings of 1998 IEEE conference on Decision and Control, Tampa, December 1998.

[5] Xiaoyi Guan and L. E. Holloway. *Assembling taskBlocks for control of discrete event systems* In preceedings of 1999 IEEE conference on Decision and Control, Phoenix, Arizona, December 1999.

[6] H.-M. Hanisch and M. Rausch, "Synthesis of supervisory controllers based on a novel representation of condition/event systems", *In proceedings of IEEE International Conference on Sstems, Man and Cybernetics*, Vancouver, October 1995.

[7] H.-M. Hanisch, A. Luder, and M. Rausch, "Controller synthesis for net condition/event system with a solution for incomplete state observation". *European Journal of Control*, (3):280-291, 1997.

[8] L. E. Holloway and S. Ranganathan. *Task blocks for synthesis of controllers for condition systems.* In Proceedings of IEEE International Conference on Systems, Man, and Cybernetics. San Diego, 1998.

[9] R.S. Sreenivas and B. H. Krogh. On condition/event systems with discrete state realizations. *Discrete Event Dynamic Theory and Applications*, 1(2), Sept. 1991.

Multiuser Receivers, Random Matrices and Free Probability *

David N.C. Tse
Department of Electrical Engineering and Computer Sciences
University of California
Berkeley, CA 94720, USA
dtse@eecs.berkeley.edu

There are often curious and deep connections between engineering and mathematics. We tell one such story here. It starts with the engineering problem of capacity analysis of direct-sequence spread spectrum wireless networks. Using results from convergence of spectra of large random matrices, we obtain simple answers for a basic model of this problem, in terms of notions of *effective bandwidths* and *effective interference.* In the process of understanding the structure of our solution better, we stumble on *free probability*, a theory for non-commutative random variables. We establish a striking connection between certain basic results in this theory and the concept of effective interference. Using free probability as a new tool, we are in turn able to analyze more complex models for the original wireless communications problem. Our story has come a full circle.

1 The Problem

With the introduction of the IS-95 Code-Division Multiple-Access (CDMA) standard, the use of spread-spectrum as a multiple-access technique in commercial wireless systems is growing rapidly in popularity. Unlike more traditional methods such as time-division multiple access (TDMA) or frequency-division multiple access (FDMA), spread-spectrum techniques are *broadband* in the sense that the entire transmission bandwidth is shared between all users at all times. This is done by the spreading of the users' signals onto a bandwidth much larger than an individual user's information rate. The advantages of spread-spectrum techniques include simpler statistical multiplexing without explicit scheduling of time or frequency slots, universal frequency reuse between cells, graceful degradation of quality near congestion, and exploitation of frequency-selective fading to avoid the harmful effects of deep fades that afflict narrowband systems.

---

*This research is supported by a National Science Foundation Early Faculty CAREER Award.

One form of spread-spectrum is *direct-sequence* CDMA. In this multiple access method, each user is assigned a *signature sequence* on which it modulates its data. A simple channel model for such a system is:

$$\mathbf{y} = \sum_{k=1}^{K} x_k \mathbf{s}_k + \mathbf{w} \in \Re^N, \tag{1}$$

where K is the number of users, x_k the transmitted data symbol of user k, $\mathbf{w} \sim N(0, \sigma^2 I_N)$ is the additive Gaussian noise in the channel, and $\mathbf{y}$ is the received vector. We assume that $E[x_k] = 0$, $E[x_k^2] = p_k$, the received power of user k. The parameter N, sometimes called the *processing gain*, is the ratio of the channel bandwidth and the data symbol rate and quantifies the amount of spreading in the system. The larger the channel bandwidth, the larger N can be. Geometrically, communication takes place in a signal space of dimension N, with each user occupying a signal direction defined by its signature sequence $\mathbf{s}_k$. The parameter N can also be thought of as the number of degrees of freedom in the system. The task of a receiver is to estimate the data symbol of each of the users from the received vector $\mathbf{y}$. We assume that the receiver has knowledge of the signature sequences and received powers of the users.

A natural class of receivers are the *linear receivers*. Focusing on user 1, the estimate $\hat{x}_1$ of the transmitted symbol x_1 is of the form $\hat{x}_1 = \mathbf{c}_1^t \mathbf{y}$, where $\mathbf{c}_1 \in \Re^N$ is the linear receiver for user 1. For example, choosing $\mathbf{c}_1 = \mathbf{s}_1$ yields the well-known *matched-filter* receiver; it projects the received signal onto the direction of the signature sequence of user 1. This is the receiver used in current-generation CDMA systems, and is simple in the sense that the receiver needs only keep track of the signature sequence of user 1 and not the others. However, one can expect that better performance can be attained by taking advantage of the knowledge of the signature sequences and received powers of other users as well.

A commonly used measure to evaluate the performance of linear receivers is the output *signal-to-interference ratio*:

$$\mathrm{SIR}_1 := \frac{\left(\mathbf{c}_1^t \mathbf{s}_1\right)^2 p_1}{(\mathbf{c}_1^t \mathbf{c}_1)\sigma^2 + \sum_{i \neq 1} \left(\mathbf{c}_1^t \mathbf{s}_i\right)^2 p_i}$$

This is simply the ratio of the variance of user 1's signal to the variance of noise plus interference from other users, measured at the output of the linear receiver. It is not too difficult to show that the optimal receiver which maximizes the output SIR is the *minimum mean-square error* (MMSE) receiver which minimizes $E[(\hat{x}_1 - x_1)^2]$. This is the classic least squares problem and has a well-known solution. The MMSE receiver and its SIR performance is given by:

$$\begin{aligned} \mathbf{c}_1 &= (SDS^t + \sigma^2 I)^{-1} \mathbf{s}_1 \\ \mathrm{SIR}_1 &= p_1 \mathbf{s}_1^t (SDS^t + \sigma^2 I)^{-1} \mathbf{s}_1 \end{aligned} \tag{2}$$

where

$$S := [\mathbf{s}_2, \ldots, \mathbf{s}_K], \qquad D := \mathrm{diag}\,(p_2, \ldots, p_K). \tag{3}$$

The MMSE receiver is an example of a *multiuser receiver* which, in contrast to the conventional matched filter, exploits the information about the other users to mitigate their interference.

The capacity analysis problem is this: given users each with its own target SIR requirement, find the mix of users that can be admissible under the MMSE receiver. This problem is currently relevant in the design of future-generation wireless systems. The capability of multiuser receivers needs to be better understood to assess whether the performance gain over the conventional receiver warrants the additional complexity in implementation.

2 The Solution

Although (2) gives a formula for the SIR of the MMSE receiver, the complicated dependence on the signature sequences and received powers of the users makes it difficult to be used directly for capacity analysis. Moreover, the performance certainly depends on which specific sequences are used. We bypass these drawbacks by instead using a *random signature sequence model.* The entries of s_i's are modeled as i.i.d. zero-mean random variables, with variance normalized to be $1/N$, and are also independent across users. This model provides analytical tractability, and it is often a reasonable approximation in practice as the wireless multipath channel typically randomizes the sequences of the users. Moreover, many CDMA systems use pseudorandom spreading sequences. It should also be noted that the random sequence model is used only for performance analysis purpose; from the point of view of the receiver, the assumption of perfect knowledge of (the realization of) the signature sequences is still retained. In practice, this information is obtained through an adaptive tracking algorithm.

In this model, SIR_1 is a random variable, being a function of the random sequences. The following is the key result, showing that in a large system, SIR_1 converges to a deterministic constant. To simplify the notation, we set $p_1 = 1$ without loss of generality.

Theorem 1 *[1] Let $N, K \to \infty$ such that $\frac{K}{N} \to \alpha$, and assume that the empirical distribution of the users' powers converges to a limiting distribution F. Then SIR_1 converges in probability to β^*, where β^* is the unique solution to the fixed-point equation:*

$$\beta^* = \frac{1}{\sigma^2 + \alpha \int_0^\infty I(p, \beta^*) dF(p)}$$

where

$$I(p, x) \equiv \frac{p}{1 + px}$$

Here, α is the system loading in terms of number of users per degree of freedom. Heuristically, the result says that in a large system,

$$\mathrm{SIR}_1 \approx \frac{1}{\sigma^2 + \frac{1}{N} \sum_{k=2}^{K} I(p_k, \mathrm{SIR}_1)}$$

Although in general this fixed-point equation has no closed-form solution, it can be shown that the fixed point can be computed numerically by simply iterating from any initial point. More importantly, if user 1 has a target SIR requirement β, then it follows from the monotonicity of the fixed-point equation that its SIR requirement is satisfied asymptotically if and only if

$$\beta \geq \frac{1}{\sigma^2 + \frac{1}{N}\sum_{k=2}^{K} I(p_k, \beta)}. \tag{4}$$

This condition can be checked explicitly.

It is natural to define $I(p, \beta)$ as the *effective interference* of an interferer with received power p on a user with a target SIR β. This summarizes succinctly the effect of an interferer. It is interesting to contrast this to the corresponding result for the matched filter receiver [1]: the effective interference under the matched filter is simply $I_{\text{mf}}(p) = p$. Note that under the matched filter, the effect of an interferer is proportional to its received power. In contrast, under the MMSE receiver, the effect is highly nonlinear in the received power, and ceiling out at $\frac{1}{\beta}$. This is a testimony to the interference suppression capability of the MMSE receiver.

We can apply this result to analyze the capacity of a power-controlled system under the MMSE receiver. Consider a set of users belonging to J classes, with target SIR requirement of β_j for users in class j. Moreover, suppose we can control the received powers of the individual users, but subject to a received power constraint of $\bar{p}_j$ for users in class j. Then it can be shown that α_j users per degree of freedom in class j can be simultaneously supportable under the MMSE receiver if and only if:

$$\sum_{j=1}^{J} \alpha_j \frac{\beta_j}{1+\beta_j} \leq \min_{1 \leq j \leq J} \left[1 - \frac{\beta_j \sigma^2}{\bar{p}_j}\right].$$

The set of class mixes $(\alpha_1, \ldots, \alpha_J)$ of users that can be simultaneously supportable is the *capacity region* of the system. The above equation says that the capacity region admits a simple description via a single linear constraint. When there are no power constraints, the right hand side simplifies to 1.

It is now natural to define the *effective bandwidth* of a user with target SIR requirement of β:

$$e(\beta) = \frac{\beta}{1+\beta}.$$

This is the fraction of a degree of freedom occupied by the user. To compute the admissibility of a set of users, one only needs to add up the effective bandwidths of the individual users. The additivity of effective bandwidths follows from the decoupling of the aggregate interference effect into the effective interference of the individual interferers.

Two curious questions arise at this point:

- Why does the SIR converge to a deterministic limit independent of the random signature sequences?

- Why does the interfering effects of the users decouple in such a simple way?

We investigate these two questions in the following sections.

3 Random Matrices

Recall that the SIR of user 1 (with unit received power) is given by:

$$\mathrm{SIR}_1 = \mathbf{s}_1^t (SDS^t + \sigma^2 I)^{-1} \mathbf{s}_1 \tag{5}$$

where S and D are defined in eqn. (3). Observe that S depends only on the signature sequences of the interferers and is hence independent of $\mathbf{s}_1$. Direct computation yields:

$$E[\mathrm{SIR}_1 | S, D] = \frac{1}{N} \mathbf{Tr} \left(SDS^t + \sigma^2 \right)^{-1}.$$

Moreover, it can be shown that the conditional variance goes to zero like $1/N$. Hence,

$$\mathrm{SIR}_1 - \frac{1}{N} \mathbf{Tr} \left(SDS^t + \sigma^2 \right)^{-1} \xrightarrow{\mathcal{P}} 0. \tag{6}$$

We can write:

$$\frac{1}{N} \mathbf{Tr} \left(SDS^t + \sigma^2 I \right)^{-1} = \int_0^\infty \frac{1}{\lambda + \sigma^2} dG_N(\lambda) \tag{7}$$

where λ has the (random) empirical eigenvalue distribution of SDS^t. Thus, the convergence of the SIR hinges on the convergence of the spectrum G_N. This problem has been solved by [2, 3], where they show that the limit exists and moreover does not depend on the distribution of the individual elements of S. The limiting spectrum is described by a functional fixed-point equation for its *Steltjes transform*. The Steltjes transform of a distribution G is defined to be:

$$m(z) := \int \frac{1}{\lambda - z} dG(\lambda)$$

The functional fixed-point equation for the Steltjes transform $m^*(z)$ of the limiting spectrum G^* of SDS^t is given by:

$$m(z) = \frac{1}{-z + \alpha \int \frac{\tau dF(\tau)}{1 + \tau m(z)}} \tag{8}$$

where F is the limiting spectrum of D.

This seems like a complicated characterization of the limiting spectrum. However, we observe from (6) and (7) that what we need to characterize the limiting SIR is precisely the Steltjes transform of the limiting spectrum of SDS^t evaluated at $z = -\sigma^2$. Theorem 1 now follows.

4 Free Probability

The effective interference interpretation follows directly from the random matrix result (8). Let us then take a deeper look at the structure of this equation from a different perspective.

Consider the scenario when there are two groups of interferers $\mathcal{C}_1$ and $\mathcal{C}_2$, one in which the interferers have common received power p and one in which the interferers have common received power q respectively. Suppose there are K_1 users in group $\mathcal{C}_1$ and K_2 users in group $\mathcal{C}_2$, with $K_1/N = \alpha_1$ and $K_2/N = \alpha_2$. The key random matrix SDS^t can be written as an outer sum:

$$SDS^t = p\sum_{i\in\mathcal{C}_1} \mathbf{s}_i\mathbf{s}_i^t + q\sum_{i\in\mathcal{C}_2} \mathbf{s}_i\mathbf{s}_i^t \equiv U_1 + U_2.$$

The asymptotic interfering effect of each group $\mathcal{C}_i$, when present in isolation, depends only on the spectrum of the random matrix U_i. How the overall interfering effect of the two groups can be decoupled into the effects of the individual groups is then a question of how the spectrum of the matrix $U_1 + U_2$ depends on the individual spectra of the matrices U_1 and U_2. This leads to a more general question: if we are given the spectra of two random matrices A and B, what can we say about the spectrum of the sum $A + B$?

For *deterministic* matrices A and B, one cannot in general determine the eigenvalues of $A + B$ from those of A and of B alone, as they depend on the eigenvectors of A and B as well. However, it turns out that for large random matrices A and B satisfying a property called *freeness*, the limiting spectrum of $A+B$ can indeed be determined from the individual spectra of A and B. This is a central result in *free probability theory*, which we very briefly introduce now. For more details, please consult [4] or [5].

Classical probability theory is concerned with commutative random variables. Free probability, on the other hand, deals with non-commutative ones. More formally, a (non-commutative) probability space $(\mathcal{A}, \varphi)$ is an algebra $\mathcal{A}$ over $\mathcal{C}$ with an unit element 1 and endowed with a linear functional, called the *trace*, $\varphi : \mathcal{A} \to \mathcal{C}$, $\varphi(1) = 1$. Elements of $\mathcal{A}$ are called (non-commutative) random variables. Classical probability theory is obtained when the algebra $\mathcal{A}$ consists of scalar random variables and the functional φ is the standard expectation operator. The focus of the theory however is on random variables which do not commute.

The distribution of a random variable $X \in \mathcal{A}$ is specified by the moments $\varphi(X^k)$, for $k \geq 1$. The distribution defined in this form allow us to compute the expectation of any function of the random variable X that can be approximated by polynomials. Similarly, the *joint distribution* of a collection of random variables $X_1, \ldots, X_m \in \mathcal{A}$ is specified by all the joint moments $\varphi(X_{i_1} \ldots X_{i_p})$, $p \geq 1$.

The central notion in classical probability theory is *independence.* In the notations introduced above, two scalar random variables X, Y are independent if $\varphi(X^kY^l) = \varphi(X^k)\varphi(Y^l)$ for all k, l. The analogous notion in free probability theory is *freeness.*

Definition 2 *Let X and Y be two random variables. Let $\mathcal{A}_1$ be the sub-algebra generated by X and the unit 1 and $\mathcal{A}_2$ be the sub-algebra generated by Y and the unit 1 respectively,* *i.e. they consist of polynomials of X and of Y respectively. The random variables X and Y are* free *if $\varphi(Z_1 \dots Z_m) = 0$ whenever $\varphi(Z_k) = 0$ for all $k = 1, \dots, m$ and $Z_k \in \mathcal{A}_{i(k)}$ where consecutive indices $i(k) \neq i(k+1)$ are distinct.*

For independent random variables, the joint distribution can be specified completely by the marginal distributions. For free random variables, the same result can be proved, directly from definition. In particular, if X and Y are free, then the moments $\varphi((X+Y)^n)$ of $X+Y$ can be completely specified by the moments of X and the moments of Y. The distribution is naturally called the *free convolution* of the two marginal distributions. Classical convolution can be computed via transforms: the log moment generating function of the distribution of $X+Y$ is the sum of the log moment generating functions of the individual distributions. For free convolution, the appropriate transform is called the R-transform. This is defined via the Steltjes transform.

Given a random variable X, let

$$m_X(z) = E\left[\frac{1}{\lambda - z}\right]$$

be the Stieltjes transform of its distribution. (The expectation is taken with respect to the distribution of X.) Let m_X^{-1} be the inverse of m_X. The R-transform is defined as:

$$R(\beta) := \frac{1}{\beta} - m_X^{-1}(\beta).$$

Theorem 3 *If X and Y are two free random variable, then $R_{A+B} = R_A + R_B$.*

A good example of non-commutative random variables are random matrices. Let $\mathcal{A} = M_N$ be the algebra of complex N by N random matrices whose entries are scalar random variables defined on some underlying common probability space. The trace is defined as:

$$\varphi_N(X) := \frac{1}{N} E\left[\mathrm{Tr} X\right].$$

For any N by N random Hermitian matrix $X \in M_N$ with random eigenvalues $\lambda_1, \dots \lambda_N$, the rth moment of X in the non-commutative probability space (M_N, φ_N) is given by

$$\varphi_N(X^r) = \frac{1}{N} E[\mathrm{Tr} X^r] = \frac{1}{N} E\left[\sum_{i=1}^{N} \lambda_i^r\right]$$

If we let $F_X(\cdot)$ be the expected empirical distribution of the eigenvalues of X:

$$F_X(\lambda) := \frac{1}{N} E\left[|\{i : \lambda_i \leq \lambda\}|\right]$$

then the moments of the distribution F_X are precisely the moments of X as a non-commutative random variable.

The deep connection between random matrices and free probability is first made by Voiculescu [6], who showed that in a lot of cases of interest, large random matrices become asymptotically free. This answers the question we posed earlier: the limiting spectra of the sum of two large random matrices is the free convolution of the limiting spectra of the individual matrices, if they become asymptotically free.

Returning to our specific problem, it can be shown as a consequence of Voiculescu's results that the matrices $U_1 = p\sum_{i\in C_i} \mathrm{s}_i\mathrm{s}_i^t$ and $U_2 = \sum_{i\in C_2} \mathrm{s}_i\mathrm{s}_i^t$ are in fact asymptotically free as $N, K \to \infty, K/N \to \alpha$. . The R-transforms can be computed explicitly for this problem:

$$R_{U_1}(\beta) = \alpha_1\frac{p}{1+p\beta}, \qquad R_{U_2}(\beta) = \alpha_2\frac{q}{1+q\beta},$$

and

$$R_{U_1+U_2}(\beta) = R_{U_1}(\beta) + R_{U_2}(\beta).$$

What is the connection of this to the notion of effective interference? Observe that the R-transform satisfies:

$$m(z) = \frac{1}{-z + R[m(z)]}.$$

Putting in $z = -\sigma^2$ and noting that $m(-\sigma^2)$ is the limiting SIR, we conclude that at target SIR requirement of β, $R_{U_i}(\beta), i = 1, 2$, is the effective interference of group C_i when considered in isolation, and $R_{U_1+U_2}(\beta)$ is the limiting aggregate interference of the two groups when both are present. **The decoupling of effective interference is nothing but the additivity of the R-transforms of free random matrices.**

5 Back to Multiuser Receivers

The previous section re-interprets the structure of the fixed-point equation in Theorem 1 in terms of notions from free probability. The proof itself does not require free probability, as the structure of the relevant random matrix SDS^t is a classical one for which a limit theorem already exists. However, the analysis of more sophisticated models [7, 8] led to non-classical random matrix structure, for which new limit theorems had to be proved. Free probability is a valuable tool for solving these problems. We discuss one such problem here [8].

One detrimental effect of a wireless link is the random *fading* of the channel due to constructive and destructive interference between multiple signal paths. A countermeasure to alleviate this problem is the use of multiple antennas at the receiver. Provided that the antennas are placed sufficiently far apart, the received signals at the antennas fade independently and the probability is high that at least one of the antennas receives a strong copy of the transmitted signal. The following is a model of a direct sequence CDMA system with 2 receive antennas:

$$\mathbf{y}(l) = \sum_{k=1}^{K} \gamma_k(l) x_k \mathrm{s}_k + \mathbf{w}(l) \in \mathcal{C}^N, \qquad l = 1, 2.$$

where as before, $x_k, \mathbf{s}_k$ are the transmitted symbol and signature sequence of the kth user respectively, $\mathbf{y}(l)$ is the received signal at antenna l, and $\gamma_k(l)$ is the random channel gain from user k to an antenna l.

As in the basic model, we can consider the MMSE estimator for transmitted symbol x_1, based now on $\mathbf{y}(1)$ and $\mathbf{y}(2)$, the received signals at both of the antennas. In place of SDS^t for the basic model, the key random matrix for this problem is $\bar{S}\bar{S}^\dagger$, where

$$\bar{S} = \begin{bmatrix} SB_1 \\ SB_2 \end{bmatrix}$$

and $S = [s_2 \dots s_k]$ and $B_l = \mathrm{diag}(\gamma_2(l), \dots, \gamma_K(l))$.

There is strong dependency between entries in the random matrix $\bar{S}$, due to the repetition of the signature sequences at the two receive antennas. Existing results on spectral convergence do not apply. However, we observe that $\bar{S}\bar{S}^\dagger$ and $\bar{S}^\dagger\bar{S}$ have the same nonzero eigenvalues, and the latter can be written as a sum of two matrices:

$$\bar{S}^\dagger\bar{S} = B_1^\dagger S^t S B_1 + B_2^\dagger S^t S B_2$$

We have the following theorem.

Theorem 4 *[1] If B_1, B_2 are independent and zero mean, then the matrices $B_1^\dagger S^t S B_1$ and $B_2^\dagger S^t S B_2$ are asymptotically free as $N, K \to \infty, K/N = \alpha$.*

Moreover, it can be shown that if S_1 and S_2 are two independent matrices, each having the same distribution as S, then $B_1^\dagger S_1^t S_1 B_1$ and $B_2^\dagger S_2^t S_2 B_2$ are also asymptotically free. The implication is that the limiting spectrum of $\bar{S}^t\bar{S}$ is the same as if independent copies of the signature sequences were received at the different antennas. For this latter "independent sequence" model, existing random matrix results can be applied and the performance of the MMSE receiver can be analyzed. More details can be found in [9] in these proceedings. The main point to be made here is that the important notion is freeness and not independence: although there is strong dependence between the entries of the matrix $\bar{S}$, the randomness in D_1 and D_2 is enough to make the components free. This result translates into the engineering conclusion that there is asymptotically no loss of degrees of freedom for communication even though the signature sequences are repeated at each of the receive antennas.

6 Conclusions

In this paper, we develop interesting connections between the engineering problem of capacity analysis of multiuser receivers on the one hand, and random matrices and free probability theory on the other. The connections are unexpected and deep. Is there anything fundamental relating the least square estimation problem and free probability, or is the connection purely coincidental? The reader is encouraged to look further.

References

[1] D. Tse and S. Hanly, "Linear multiuser receivers: Effective interference, effective bandwidth and capacity," *IEEE Trans. Inform. Theory*, vol. 45, pp. 641–657, Mar. 1999.

[2] V. A. Marcenko and L. A. Pastur, "Distribution of eigenvalues for some sets of random matrices," *Math. USSR-Sb.*, vol. 1, pp. 457–483, 1967.

[3] J. W. Silverstein and Z. D. Bai, "On the empirical distribution of eigenvalues of a class of large dimensional random matrices," *Journal of Multivariate Analysis*, vol. 54, no. 2, pp. 175–192, 1995.

[4] D. V. Voiculescu, K. J. Dykema, and A. Nica, *Free Random Variables.* CRM Monogragh Series, Volume 1, Providence, Rhode Island, USA: American Mathematical Society, 1992.

[5] D. Voiculescu, "Lectures on free probability theory," 1998. Notes for a Course at the Saint-Flour Summer School on Probability Theory (unpublished).

[6] D. Voiculescu, "Limit laws for random matrices and free products," *Inventiones mathematicae*, vol. 104, pp. 201–220, 1991.

[7] J. Evans and D. Tse, "Linear multiuser receivers for multipath fading channels," *Submitted to IEEE Trans. Inform. Theory*, 1999.

[8] S. Hanly and D. Tse, "Resource pooling and effective bandwidths in CDMA systems with multiuser receivers and spatial diversity," *Submitted to IEEE Trans. Inform. Theory*, 1999.

[9] S. Hanly and D. Tse, "Resource pooling and effective bandwidths for CDMA antenna arrays," in *Proc. Allerton Conference on Communications, Computing and Control, Monticello, Illinois*, Sept. 1999.

Stability properties of incremental redundancy in CDMA packet data networks[*]

Rajiv Vijayakumar and Kimberly M. Wasserman
EECS Department, University of Michigan, Ann Arbor
`{rvijayak,wass}@eecs.umich.edu`

Abstract

This paper focuses on the study of schemes that improve the throughput performance of a conventional DS-CDMA packet data network, and in particular, eliminate, or drastically mitigate, bistability that arises from interference. Two schemes are considered: spreading gain control (SGC) and incremental redundancy (IR). Under SGC, the spreading gains of the users are varied in response to changing interference conditions. Under IR, when a packet with uncorrectable errors is received, it is not discarded but combined with retransmitted packets via bit-by-bit addition of the corresponding matched filter outputs. The IR scheme is related to SGC at some level, but requires no estimation of the interference. It is shown that the SGC system possesses a single equilibrium. The IR system may possess multiple equilibria. Numerical results show, however, that the equilibria are typically close together toward the origin. It is shown that, if there is only a single equilibrium, then under certain conditions, there exists a time T such that, for a IR system with a sufficiently large number of users, the system is within any given neighborhood of this equilibrium point with high probability for all times greater than T. An analogous result holds for SGC.

1 Introduction

Just as in narrowband-ALOHA and CSMA/CD, the throughput performance of spread spectrum wireless packet data networks, Direct-Sequence Code Division Multiple Access (DS-CDMA) as well as Frequency-Hopped CDMA, can be severely hampered due to saturation effects caused by inherent bistable behavior [1] [2]. Bistable systems possess three points of equilibria: A desirable stable (operating) point at which throughput is high and delay is low; an unstable point; and an undesirable stable (saturation) point at which throughput is low and delay is high [3]. In steady-state, the network essentially operates at the two stable points with occasional transitions from one point to the other. Once the system drifts to the undesirable stable point, it typically remains there for long periods of time, which accounts for the overall poor performance.

Up-and-coming third-generation mobile networks [4] [5] [6], which use wideband DS-CDMA technology, are expected to provide packet-mode data communication services in addition to voice, video, and multimedia services. Given this, it is of interest to design algorithms or schemes that eliminate, or at least significantly mitigate, bistability in DS-CDMA packet

---

[*] This work was supported in part by the U.S. Army Research Office under grants DAAH04-96-1-0377 and DAAH04-96-1-0177.

data networks. In this paper, we study two such schemes: spreading gain control (SGC) and incremental redundancy (IR). Under SGC, the spreading gains of the active users are varied in response to changing channel (interference) conditions, that is, number of active users, which results in drastic improvements in throughput [7]. The IR scheme is a simple form of diversity or code combining, where corrupted packets are *not* discarded but combined in some fashion with retransmitted packets. In particular, the matched filter outputs (soft decision) for the original and retransmitted packets are added bit-by-bit before making a decision. This scheme is related, at some level, to spreading gain control, but requires *no* estimation of the number of active users.

This and other forms of code combining have been applied to DS-CDMA systems in [8] [9] [10] [11] and [12]. These studies show that code combining achieves high performance by integrating channel coding with feedback control. However, it is difficult to analyze the dynamics of these systems as they are multidimensional and possess memory. Indeed, the packet error probability, which determines the throughput and delay performance, as well as the stability properties, depends on the current number of active users, in addition to the number of active users in previous slots. In past works, approximate systems have been studied. For example, in [8], the number of active users in each slot are independent and identically distributed Poisson random variables. In [9] and [10], the number of active users is fixed.

In [11], two-dimensional systems that upper and lower bound the performance of the actual system are studied. In [12], the equilibrium points of a multidimensional code combining system are obtained, and their stability properties are discussed using a definition of stability based on one-dimensional systems. In this paper, we study the dynamics and stability properties of the exact DS-CDMA system under SGC, which is one-dimensional, and IR, which is multidimensional. We show that under SGC, the system possesses a single equilibrium. Under IR, we show that the system may possess multiple equilibria. Numerical results show, however, that the asymptotically stable equilibria are typically close together toward the origin. We also show that, if there is only a single equilibrium, then under certain conditions, there exists a time T such that, for a IR system with a sufficiently large number of users, the system is within any given neighborhood of this equilibrium point with high probability for all times greater than T. An analogous result holds for SGC.

The paper is organized as follows. In Section 2, we describe the DS-CDMA system. In Section 3, we study SGC, and in Section 4, we study IR.

2 DS-CDMA

We consider a DS-CDMA data network consisting of a single base station and M mobile wireless users. The users generate non-real-time packet data and transmit over a common wideband communication channel to the base station using different spreading codes. Each user transmits with a spreading gain of $N \geq 1$ chips per bit and perfectly controls its transmission power to deliver a constant received signal power of P_r at the base station. The time-axis is divided into contiguous equal-length time slots, each of which has a duration equal to the time required to transmit a single packet. The users initiate transmissions only at the beginning of time slots, and spread their signals across the entire available spectrum, that is, the chip rate of each user is equal to $L\,N$ chips per slot, where L is the number of bits per packet.

Each user has a buffer of size one. If the user has a packet in its buffer, it is considered "backlogged;" otherwise, it is "idle." If a user transmits during a slot, it is "active" in that slot; backlogged users are active with probability *one*. An idle user generates a packet with probability $0 \leq \lambda \leq 1$ per slot independent of M. The receivers at the base station are assumed to be of the conventional matched filter type. The $\mathcal{E}_b/I_0$ experienced by an active user then becomes a function of the spreading gain, the number of other active users, and the lump-sum background interference and thermal noise power. Specifically,

$$\mathcal{E}_b/I_0 = \frac{N}{(j-1)+\eta}, \tag{1}$$

where $j \in \{1, 2, \ldots M\}$ is the number of active users and $\eta = \sigma^2/P_r$. The probability $p : \mathbb{R}_+ \rightarrow [0,1]$ of a successful packet transmission by an active user is a non-decreasing function of the $\mathcal{E}_b/I_0$ experienced by the user throughout the packet transmission time; the exact functional form of p depends on the choice of modulation technique, channel code, interleaving depth, etc. The outcomes of the transmission attempts of the active users are mutually independent.

We are interested in the behavior of DS-CDMA data networks with a large number of users. When a large number of users are backlogged, the $\mathcal{E}_b/I_0$ experienced by a user is dominated by the contribution due to the interference. In the sequel, we will assume that the effects of thermal noise are negligible and take $\eta = 0$. In order to preserve bistability for arbitrarily large M, N is increased linearly with M. Intuitively, if λ is constant, then the net arrival rate $M\lambda$ into the system when it is empty grows with M, and therefore we need increased bandwidth (large N) in order to maintain the throughput level.

Define the random sequence $X = \{X_n; n \in \mathbb{Z}_+\}$, where $X_n \in [0,1]$ is the fraction of backlogged users at the beginning of the n^{th} time slot. The state of the system at time n is completely described by X_n, $n \in \mathbb{Z}_+$. Let $N = \gamma M$ for some constant $\gamma > 0$. Then the arrival rate when the state is x is $\lambda M(1-x)$, and the departure rate (throughput or goodput) is

$$Mxp\left(\frac{N}{Mx-1}\right) = Mxp\left(\frac{\gamma}{x-\frac{1}{M}}\right),$$

for large enough M, $0 \leq x \leq 1$.

A state $x \in [0,1]$ is defined to be an equilibrium state of the system if $E[X_{n+1} \mid X_n = x] = x$. Then for an equilibrium state $x \in [0,1]$, we must have

$$x\left(1 - p\left(\frac{\gamma}{x-\frac{1}{M}}\right)\right) + \lambda(1-x) = x.$$

Thus for an arrival rate λ, the equilibrium states are given by the solutions of the equation

$$\lambda = \frac{xp\left(\frac{\gamma}{x-\frac{1}{M}}\right)}{1-x},$$

$0 \leq x \leq 1$.

As is the case in many systems, whether or not multiple equilibria exist in a DS-CDMA data network depends on the specific values of the parameters. In particular, it is always possible to select a fixed spreading gain large enough to result in a single equilibrium state. However, this static approach may lead to low throughput and inefficient bandwidth utilization (low spectral efficiency) since a large spreading gain is not necessary when the number of backlogged or active users is low. This suggests an approach by which the spreading gain of each backlogged user is controlled dynamically as a function of the number of active users (interference). This is the approach adopted in SGC [7], where the optimal spreading gain is shown to be proportional to the number of active users.

3 Spreading gain control (SGC)

The basic idea behind dynamic spreading gain control is to (in)decrease the spreading gain of the users as the interference level (in)decreases, which can result in *drastic* improvements in throughput compared to fixed spreading gains [7]. The spreading gain of a user is controlled by dynamically increasing or decreasing the time duration of a transmitted bit (decreasing or increasing the data rate of the user), while maintaining a constant chip rate. From (1), a target $\mathcal{E}_b/I_0$ can be met with a relatively small spreading gain if the number of backlogged users is small. In this case, the bit transmission duration decreases, thereby increasing the throughput.

As discussed in [7], define γ^* by $\gamma^* = \arg\max_{\gamma\geq 1} (p(\gamma)/\gamma)$ and control the spreading gain so that $N = \gamma^*(Mx - 1)$. Then, if the system is in state x, the departure rate is $Mxp(\gamma^*)$ and therefore for an equilibrium state x, we have

$$M\lambda(1-x) = Mxp(\gamma^*),$$

so that the equilibrium sates are given by the solutions of the equation

$$\lambda = \frac{x\,p(\gamma^*)}{1-x}. \tag{2}$$

Since $x/(1-x)$ is an increasing function of x, (2) shows that a system operating under SGC has a unique equilibrium state for each arrival rate $\lambda \in [0, 1]$. However, SGC requires the mobile to know the number of active users. In the next section, we will consider an alternative approach based on diversity combining that requires no estimation of the number of active users.

4 Incremental redundancy (IR)

In a system operating under IR, if the base station detects an error in a packet, it requests a retransmission but does not discard the original packet. Instead, when the packet is retransmitted, the receiver views the original and retransmitted packets as being two versions of the same packet received over different channels and uses diversity combining to decode the packet. We assume the following very simple form of diversity combining: The matched filter outputs (soft decision) for the original transmission and the retransmission are *added* (componentwise) before making a decision. This has the effect of increasing the $\mathcal{E}_b/I_0$ on successive retransmissions, and quickly adapts the realized data rate to the channel conditions using a *fixed* transmission rate. In addition, no estimation of the number of active users is needed.

4.1 Motivation

In order to motivate our interest in IR, we first point out its relation to SGC. Consider a packet that a mobile has attempted to transmit j times and let m_i be the number of active users on the i^{th} transmission attempt, $i = 1, 2, \ldots, j$, $j \in \mathbb{Z}_+$. The effective $\mathcal{E}_b/I_0$ at the receiver after combining j copies of the packet in the manner described above is then given by

$$\mathcal{E}_b/{I_0}_{\text{eff}} = \frac{jN}{m_{av} - 1}, \tag{3}$$

where

$$m_{av} = \frac{m_1 + m_2 + \ldots + m_j}{j}$$

is the average number of active users over the first j transmission attempts, $j \in \mathbb{Z}_+$. Thus if the number of active users is constant over all j transmission attempts, then for purposes of computing the probability of a successful transmission on the j^{th} attempt, the packet has effectively used a spreading gain of $j \times N$. Thus, IR can roughly be viewed as a form of SGC, but with two important differences: (i) under IR, packets effectively attempt transmission with lower spreading gains before using higher ones, and (ii) under IR, *no* information about the number of active users in a slot is required. Effectively, IR increases the spreading gain *only for packets which require it*; that is, it adapts the data rate to the channel conditions automatically. On the basis of the above observations, it seems reasonable that, in some sense, IR should perform comparably to SGC.

4.2 State description

Unlike conventional DS-CDMA or SGC, the state of an IR system is multi-dimensional, and cannot be completely described by the number of backlogged users. However, we will show that the equilibrium states for an IR system are, in fact, completely characterized by the number of backlogged users in the system.

We assume that there is some maximum number E of transmission attempts per packet; that is, if a packet is not successfully transmitted on E attempts, then it is discarded. Further retransmissions, if required, are assumed to be controlled by higher layer protocols, such as TCP. Define the random sequence $X = \{X_n; n \in \mathbb{Z}_+\}$, where

$$X_n = (x_n^1, h_n^1, x_n^2, h_n^2, \ldots, x_n^E, h_n^E),$$

and $x_n^j \in [0,1]$ is the fraction of backlogged users transmitting in slot n for the j^{th} time, and

$$h_n^j = \begin{cases} \sum_{i=1}^{j-1} \sum_{k=1}^{E} x_{n-i}^k & j > 1 \\ 0 & j = 1 \end{cases}$$

represents the transmission history of the packets transmitting for the j^{th} time, that is, the total number of active users seen in the previous $j-1$ transmissions. This history is the same for all packets transmitting for the j^{th} time, since all backlogged users transmit in every slot, $j \in \{1, 2, \ldots, E\}$. The state of the system at slot n is completely described by X_n, $n \in \mathbb{Z}_+$. Note that since all quantities are expressed as a fraction of the total number of users, the process X is implicitly indexed by M, the total number of users. This index has been suppressed to simplify notation.

It is easy to see that X forms a Markov chain for which we now derive the one-step transition probabilities. For notational convenience, we will denote a typical state of the system by $\boldsymbol{x}$, where $\boldsymbol{x} = (x^1, h^1, \ldots, x^E, h^E)$. Write $x = x^1 + \cdots + x^E$ for the total fraction of backlogged users, and make the same statistical assumptions as in Section 2. Let $p_j(x, h^j)$ denote the probability of packet success for packets transmitting for the j^{th} time when the system is in state $\boldsymbol{x}$. From (3) we have

$$p_j(x, h^j) = p\left(\frac{jN}{\frac{M(h^j+x)}{j} - 1}\right) = p\left(\frac{j\gamma}{\frac{(h^j+x)}{j} - \frac{1}{M}}\right), \tag{4}$$

$0 \le x \le 1$, $j = 1, 2, \ldots, E$. Here $\gamma = N/M$, as in Section 2. The transition probability $P\{X_{n+1} = \hat{\boldsymbol{x}} \mid X_n = \boldsymbol{x}\}$ is given by

$$P\{X_{n+1} = \hat{\boldsymbol{x}} \mid X_n = \boldsymbol{x}\} = \prod_{j=1}^{E} P\{x_{n+1}^j = \hat{x}^j \mid X_n = \boldsymbol{x}\},$$

provided $\hat{h}^1 = 0$ and $\hat{h}^j = h^{j-1} + x$ for $j = 2, \ldots, E$, and is zero otherwise. Given $X_n = x$, $\hat{x}^j$ takes value k/M, $k = 0, \ldots, Mx^{j-1}$, with probability

$$\binom{Mx^{j-1}}{Mx^{j-1} - k} \left(p_{j-1}(x, h^{j-1})\right)^{Mx^{j-1}-k} \left(1 - p_{j-1}(x, h^{j-1})\right)^k,$$

$j = 2, \ldots, E$, and $\hat{x}^1$ takes values k/M, $k = 0, \ldots, M(1-x)$ with probability

$$\binom{M(1-x)}{k} \lambda^k (1-\lambda)^{M(1-x)-k}.$$

4.3 Dynamics

From the transition probabilities above, we have $E[X_{n+1} \mid X_n = x] = \hat{x}$, where

$$\hat{x}^1 = \lambda(1-x)$$
$$\hat{h}^1 = 0,$$

and for $k = 2, 3, \ldots, E$,

$$\hat{x}^k = x^{k-1}(1 - p_{k-1}(x, h^{k-1}))$$
$$\hat{h}^k = h^{k-1} + x, \tag{5}$$

$0 \leq x \leq 1$.

Define

$$\tilde{p}_j(x, h^j) = p\left(\frac{j\gamma}{\frac{(h^j+x)}{j}}\right).$$

If p is continuous, then $\tilde{p}_j(x, h^j)$ is the limit as $M \to \infty$ of $p_j(x, h^j)$. Let g denote the function which maps a state x of the system to $\tilde{x} = g(x)$ by $\tilde{x}^1 = \lambda(1-x)$ and $\tilde{h}^1 = 0$, and for $k = 2, 3, \ldots, E$, $\tilde{x}^k = x^{k-1}(1 - \tilde{p}_{k-1}(x, h^{k-1}))$ and $\tilde{h}^k = h^{k-1} + x$, $0 \leq x \leq 1$.

Given any state x of an IR system, define the *deterministic* sequence $Y_x = \{Y_{n;x}; n \in \mathbb{Z}_+\}$ by $Y_{0;x} = x$ and $Y_{n+1;x} = g(Y_{n;x})$. The process Y is, in a sense, the limiting process as $M \to \infty$ of the stochastic process X; this notion is made precise in the two theorems below.

Suppose that the function g defined above is Lipschitz continuous on the state space. Let $\|z\|$ denote the Euclidean norm of z. We then have the following theorem (see [13]).

Theorem 1 *Under the above assumption, for any $t \in \mathbb{Z}_+$, and any $\delta > 0$,*

$$\max_{0 \leq n \leq t} Pr\{\|X_n - Y_{n;X_0}\| > \delta\} \xrightarrow{M \to \infty} 0.$$

The above theorem implies that for any $n \in \mathbb{Z}_+$, we can pick M sufficiently large so that, at time n, the stochastic system X is, with high probability, very near the deterministic system Y.

Now suppose further that the process $Y_{n;\cdot}$ has a unique equilibrium; that is, there exists a unique x^* such that $Y_{n;x^*} = x^*$ for all n. Also suppose that starting from any initial state x, the process $Y_{n;x}$ converges as $n \to \infty$ to x^* (*i.e.*, asymptotic stability on the state space). We then have the following theorem (see [13]).

Theorem 2 *Under the above assumptions, for any $\delta > 0$,*

$$\sup_{n\geq 0} Pr\{\|X_n - Y_{n;X_0}\| > \delta\} \xrightarrow{M\to\infty} 0.$$

This implies (see [13]) that there is a time T such that for all sufficiently large M the system is very near the equilibrium state with high probability for *all* times greater than T.

Remark. It is possible to define a limiting deterministic system analogous to Y for SGC. The unique equilibrium point y^* for this system is given by the unique solution to (2). Further, it is easily established that this equilibrium is asymptotically stable on the state space; for instance, by applying the Lyapunov function (see [14]) $V(y) = |y - y^*|$. Consequently, a result analogous to Theorem 2 always holds for a SGC system.

We now characterize the equilibrium points for an IR system. If $\mathbf{x}$ is an equilibrium state, we must have $\hat{\mathbf{x}} = \mathbf{x}$ in (5), that is, $x^1 = \lambda(1-x)$ and $h^1 = 0$, and for $k = 2, \ldots, E$, $x^k = x^{k-1}(1 - p_{k-1}(x, h^{k-1}))$ and $h^k = h^{k-1} + x$, and finally, $x = x^1 + x^2 \cdots + x^E$, $0 \leq x \leq 1$. Solving this system of equations yields

$$h^j = (j-1)x$$
$$x = \lambda(1-x)\left(1 + \sum_{j=1}^{E-1}\prod_{k=1}^{j}(1 - p_k(x, h^k))\right),$$

$0 \leq x \leq 1$, $j = 1, 2, \ldots, E$. Thus, although the state space is multi-dimensional, for a given arrival rate λ, the equilibrium states $\mathbf{x}$ are *uniquely* determined by the *number* x of backlogged users in those states. Further, given λ, the values of x corresponding to the equilibrium states are given by the roots of the equation

$$\lambda = \frac{x}{(1-x)\left(1 + \sum_{j=1}^{E-1}\prod_{k=1}^{j}\left(1 - p_k\left(x, (k-1)x\right)\right)\right)}, \tag{6}$$

$0 \leq x \leq 1$.

It is easy to see that if we replace p_k in (6) by $\tilde{p}_k$, then the roots of (6) are the equilibrium points for the process $Y_{n;\cdot}$. In order for Theorem 2 to hold, there must exist a unique solution to (6) which is also asymptotically stable on the state space. We investigate asymptotic stability of the equilibria in the next section and uniqueness of the solution to (6) in section 4.5.

4.4 Asymptotic Stability

In this section, we obtain a condition via linearization under which an equilibrium is locally asymptotically stable. In the following, we write $\mathbf{y}_n$ for $Y_{n;\mathbf{x}}$, dropping the dependence on the initial state $\mathbf{x}$ from the notation. Let $\mathbf{y}^*$ denote an equilibrium state for $Y_{n,\cdot}$. Since $\mathbf{y}^*$ is completely specified by y^*, it follows readily that $\mathbf{y}_n$ converges to $\mathbf{y}^*$ if and only if y_n converges to y^*. Let $\tilde{q} = 1 - \tilde{p}$. Then, by recursive substitution in the equation $\mathbf{y}_n = g(\mathbf{y}_{n-1})$, we obtain

$$y_n = \sum_{k=1}^{E}\lambda(1 - y_{n-k})\prod_{j=1}^{k-1}\tilde{q}\left(\frac{j^2\gamma}{\sum_{i=1}^{j} y_{n-k+i}}\right), \tag{7}$$

where, by convention, the empty product equals one.

Let $\bar{y}_n$ denote the E-vector $(y_n, y_{n-1}, \ldots, y_{n-E+1})^T$, and let $\bar{y}^*$ denote the E-vector $(y^*, y^*, \ldots, y^*$ Assume $\tilde{q}$ is everywhere differentiable. Let

$$\alpha_m \triangleq \left.\frac{\partial y_n}{\partial y_{n-m}}\right|_{\bar{y}_{n-1}=\bar{y}^*}$$

$$= \sum_{k=m+1}^{E} \lambda(1-y^*) \sum_{j=k-m}^{k-1} \frac{-\gamma}{y^{*2}} \tilde{q}'\left(\frac{j\gamma}{y^*}\right) \prod_{l=1,l\neq j}^{k-1} \tilde{q}\left(\frac{l\gamma}{y^*}\right) - \lambda \prod_{j=1}^{m-1} \tilde{q}\left(\frac{j\gamma}{y^*}\right),$$

$$m = 1, \ldots, E,$$

where $\tilde{q}'$ denotes the derivative of $q(\cdot)$, and let

$$A = \begin{bmatrix} \alpha_1 & \alpha_2 & \ldots & \alpha_{E-1} & \alpha_E \\ 1 & 0 & \ldots & 0 & 0 \\ 0 & 1 & \ldots & 0 & 0 \\ \vdots & \vdots & & \vdots & \vdots \\ 0 & 0 & \ldots & 1 & 0 \end{bmatrix}.$$

Then $\bar{y}_n = A(\bar{y}_{n-1} - \bar{y}^*) + \bar{y}^* + (r(\bar{y}_{n-1}), 0, \ldots, 0)^T$, where $\lim_{\bar{y}_{n-1}\to\bar{y}^*} \frac{|r(\bar{y}_{n-1})|}{\|\bar{y}_{n-1}\|} = 0$.

The equilibrium point $\boldsymbol{y}^*$ of Y_{n}, is asymptotically stable if the matrix A is stable, that is, if all of the eigenvalues of A lie inside the unit circle [14]. Since A is in the form of a companion matrix, this is equivalent to the condition that the roots of the equation

$$\beta^E - \alpha_1\beta^{E-1} - \alpha_2\beta^{E-2} - \cdots - \alpha_{E-1}\beta - \alpha_E = 0$$

all have modulus less than one.

4.5 Equilibrium states and throughput

For notational convenience we write $p_k(x)$ for $p_k(x, (k-1)x)$ and $f(x)$ for $(1 + \sum_{j=1}^{E-1} \prod_{k=1}^{j} (1 - p_k(x)))$. Also, let $\Lambda(x) = x/((1-x)f(x))$. We will assume that $p_k(x)$ is a differentiable function of x. Then equation (6) may be written as $\lambda = \Lambda(x)$.

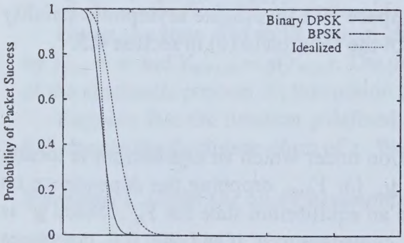


Figure 1: Probability of successful packet transmission as a function of the fraction of users backlogged for various schemes.

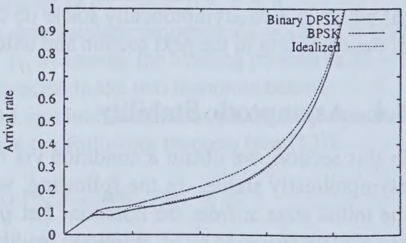


Figure 2: $\Lambda(x)$ for various schemes.

Figure 1 plots $p_1(x)$ for BPSK and binary DPSK with a packet length of 255 bits and a spreading gain-to-number of users (γ) ratio of 0.4. Figure 2 plots the corresponding $\Lambda(x)$

functions for the two cases, for $E = 5$. The idealized case plotted in these figures will be discussed below. Note that the BPSK system has an unique equilibrium state for any $\lambda \in [0, 1]$. The DPSK system has multiple equilibria for $\lambda \in [0.0862, 0.0874]$ and a single equilibrium point for all other arrival rates. The maximum difference between two equilibrium points for the same arrival rate occurs for an arrival rate of $\lambda \approxeq 0.0862$. In this case, the difference in the fraction of backlogged nodes at the two equilibrium states is 0.032 and the difference in throughput is 0.003. Thus, despite having multiple equilibria, the throughput of the system is not very different at the distinct equilibrium points.

For any $\lambda \in [0, 1]$, the existence of a solution $x \in [0, 1)$ to the equation $\lambda = \Lambda(x)$ is guaranteed by the fact that $\Lambda(x) \geq 0$, $\Lambda(0) = 0$ and $\lim_{x \to 1} \Lambda(x) = \infty$. In order for this solution to be unique for all λ, it is necessary and sufficient that $\Lambda(x)$ be a monotonically increasing function on $\Lambda^{-1}\{[0, 1]\} \cap [0, 1]$. From the expression for Λ, $\Lambda' > 0$ if and only if

$$f'(x) < \frac{f(x)}{x(1-x)} = \frac{1}{\Lambda(x)(1-x)^2}. \tag{8}$$

Since $\lambda \leq 1$ and $f(x) \geq 1$, a sufficient condition for an IR system to have a single point of equilibrium is

$$f'(x) < \min\left\{\frac{1}{(1-x)^2}, \frac{1}{x(1-x)}\right\}. \tag{9}$$

Now consider an example where the above condition does not hold. Suppose $E = 2$ and for some $0 < c < 1$, $p_1(x) = \mathbf{1}_{\{x \leq c\}}$, where $\mathbf{1}_A$ denotes the indicator function of the event A. A simple calculation then shows (see [13]) that in this extreme case, the system has multiple equilibria for $\lambda \in (c/(2(1-c)), c/(1-c)]$. Intuitively, the system has two distinct modes of operation: One in which all packets are successful in the first transmission attempt and the other in which none are successful in the first attempt.

For practical modulation schemes, $p_1(x)$ is typically close to 1 for small x, then drops sharply and stays close to zero. An idealized function $p_1(x)$ is shown in Figure 1. This function is defined by $p_1(x) = 1$ for $0 \leq x < x_1$, $1 - \frac{x - x_1}{x_2 - x_1}$ for $x_1 \leq x < x_2$, and 0 for $x_2 \leq x \leq 1$. Figure 1 plots this function for $x_1 = 0.1$ and $x_2 = 0.14$. Figure 2 plots the corresponding $\Lambda(x)$ function, where we have taken $p_k(x) = p_1(x/k)$. We can explain the behavior of $\Lambda(x)$ as follows. For $x < 2x_1$, $p_2(x) = 1$ and so we are essentially reduced to the $E = 2$ case discussed earlier. Thus, even in the general case, for small values of x, we effectively have $E = 2$. Since $x_2 < 2x_1$ and $2x_2 < 3x_1$, we have $f(x) = 2 + (1 - p_2(x))$ for $2x_1 = 0.2 \leq x \leq .28 = 2x_2$. In this region $f'(x) = -p_2'(x) = -\frac{1}{2}p_1'(x/2) = \frac{1}{2}f'(x/2)$. This demonstrates another typical feature: $f'(x)$ tends to decrease with x. Since $f(x)$ increases with x and $x(1-x)$ decreases for $x > 1/2$, we expect that the condition (8) will be satisfied for all sufficiently large x. In addition, as x increases, we see fewer and smaller "bumps" in $\Lambda(x)$.

References

[1] D. Raychaudhuri, "Performance analysis of random access packet-switched code division multiple access systems," *IEEE Trans. Commun.*, vol. 29, no. 6, pp. 895–901, 1981.

[2] A. Polydoros and J. Silvester, "Slotted random access spread-spectrum networks: An analytical framework," *IEEE J. Select. Areas Commun.*, vol. 5, no. 6, pp. 989–1002, 1987.

[3] D. Bertsekas and R. Gallager, *Data Networks*. Prentice-Hall, 2nd ed., 1992.

[4] T. Ojanperä and R. Prasad, "An overview of air interface multiple access for UMTS/IMT-2000," *IEEE Commun. Mag.*, vol. 36, no. 9, pp. 82–95, 1998.

[5] F. Adachi, M. Sawahashi, and H. Suda, "Wideband DS-CDMA for next-generation mobile communication systems," *IEEE Commun. Mag.*, vol. 36, no. 9, pp. 56–69, 1998.

[6] E. Dahlman, B. Gudmundson, M. Nilsson, and J. Sköld, "An overview of air interface multiple access for UMTS/IMT-2000," *IEEE Commun. Mag.*, vol. 36, no. 9, pp. 70–81, 1998.

[7] S.-J. Oh and K. M. Wasserman, "Dynamic spreading gain control in multi-service CDMA networks," *IEEE J. Select. Areas Commun.*, vol. 17, no. 5, pp. 918–927, 1999.

[8] A. M. Y. Bigloo, T. A. Gulliver, and V. K. Bhargava, "Maximum-likelihood decoding and code combining for DS/SSMA slotted ALOHA," *IEEE Trans. Commun.*, vol. 45, no. 12, pp. 1602–1612, 1997.

[9] S. Souissi and S. B. Wicker, "Dynamic performance of a random access, code combining CDMA system," in *Proc. 6th Internat. Symp. on Person., Indoor and Mobile Radio Commun.* (PIMRC'95), Toronto, Canada, 1995. vol. 44, pp. 304–312, 1995.

[10] S. Souissi and S. B. Wicker, "A diversity combining DS-CDMA system with convolutional encoding and Viterbi decoding," *IEEE Trans. Veh. Tech.*, vol. 44, pp. 304–312, 1995.

[11] Q. Zhang, T. F. Wong, and J. S. Lehnert, "Performance of a type-II hybrid ARQ protocol in slotted DS-SSMA packet radio systems," *IEEE Trans. Commun.*, vol. 47, no. 2, pp. 281–289, 1999.

[12] Q. Zhang, T. F. Wong, and J. S. Lehnert, "Stability of a type-II hybrid ARQ protocol for DS-SSMA packet radio systems," in *Proc. IEEE INFOCOM'98*, San Francisco, CA, vol. 3, pp. 1301-1308, Apr. 1998.

[13] R. Vijayakumar and K. M. Wasserman, "On stability of DS-CDMA wireless data networks," Tech. Rep. CGR 99-02, EECS Dept., University of Michigan, 1999.

[14] J. P. LaSalle, *The Stability and Control of Discrete Processes*. Springer-Verlag New York Inc., 1986.

Pricing Priority Classes in a Differentiated Services Network

Peter Marbach
Center for Communication Systems Research
University of Cambridge
10 Downing Street
Cambridge, CB2 3DS, UK
pm259@ccsr.cam.ac.uk

Abstract

We study the role of pricing in packet-based telecommunication networks which use priorities to provide differentiated Quality-of-Service (QoS). Customers are given the freedom to choose priorities, but are charged accordingly. Using a game theoretic framework, we study the case where customers choose an allocation of priorities to packets in order to optimize their net benefit. For the single link case, we show that there always exists a (Nash) equilibrium for the corresponding non-cooperative game, which, however, is not necessarily unique. Furthermore, we show that under each equilibrium, packet loss depends only on pricing decisions and therefore pricing becomes a tool to control link congestion.

1 Introduction

There have been proposals to use priorities to provide Quality of Service (QoS) in packet-based telecommunications networks such as the Internet (see Diffserv proposal of the IETF [4] and LAN protocol of the IEEE [7]). The basic idea is the following. Consider a network which offers a finite number of different priority levels. At each node, packets are served according to a priority rule, so that packets with high priorities will experience shorter delays and are less likely to be dropped. This mechanism enables the provision of differentiated QoS: customers with tight QoS requirements (such as IP-Telephony connections) may use high priorities, whereas customers with QoS-insensitive traffic (such as Email) use low priorities.

A question that arises in this context is how the network provider can prevent customers from marking all their packets with the highest priority. One possible solution to this problem is to implement a network controller which assigns priorities to customers based on some predefined policy. This approach gives the network tight control over the use of the priorities, but is expensive to implement. In addition, it may be difficult for the the network controller to obtain the necessary information regarding the QoS requirements and traffic patterns of the individual customers to make priority assignments in an efficient way. In an alternative approach, pricing could be employed to regulate the use of priorities. In this scenario, customers are free to choose the priorities they attach to their packets, but are charged accordingly (of course, the price for higher priorities will be higher). Leaving the issue of how to do billing and accounting aside, this scheme can easily be implemented. Furthermore, it gives customers the freedom to choose priorities that match best their service requirements. One would expect that customers will then try to meet their QoS requirements at the lowest possible cost and use high priorities only for packets with tight QoS requirements.

The goal of this paper is to study the priority service, combined with pricing, described above; we restrict our analysis to the the single link case. We adopt an economic framework where customers and their service requirements are characterized by an utility function. Roughly, submitting more high priority packets will increase the utility of an individual customer. However, doing so will also increase cost, i.e. the price the customer has to pay for using the link. It is then natural to assume that that customers will choose an allocation of packets to priorities in order to maximize their own net benefit, i.e. their utility minus cost. We model this situation as a non-cooperative game for which we wish to determine whether there exists a (Nash) equilibrium and how pricing decision influence the characteristic of an equilibrium. Due to space constraints, we will state results without proof.

Below we provide a few pointers to related work. The framework that we consider here is closely related to the Paris Metro Pricing (PMP) proposal [5] for providing QoS in the Internet. PMP partitions the network into several logically separated channels where channels differ in the prices paid for using them. This can be implemented by using the priority scheme, combined with pricing, described above. However, [5] does not analyze the existence of an equilibrium and the role of pricing. [2] provides simulation results to demonstrate that priorities can be an efficient tool for providing QoS in packet-based networks. In particular, [2] illustrates that using priorities, combined with pricing, achieves a higher a social welfare than a best-effort/flat-rate scheme. Gupta et. al. investigate in [3] a priority service similar to the one propose here, however they consider the situation where the network dynamically changes the prices associated with the different priority classes in order to track a socially optimal allocation. This approach is theoretically appealing, but expensive to implement. The scheme considered in this paper is simpler as prices associated with the different service classes are fixed. Park et. al. also consider in [1, 6] a service where packets get marked according to customer's QoS requirements. However, there is no price associated with the different traffic classes, i.e. the costs incurred to customers are purely performance related. They study the situation where users are free to choose their traffic class [6], as well as where users indicate their QoS requirements and a network controller assigns network resources [1].

2 Link Model

For our analysis, we assume that time is divided into slots and consider a single link with a capacity of C packets per time slot. There is no buffer available and packets that do not get transmitted in a given time slot are dropped. The link supports a finite set $I = \{1, ..., N\}$ of different priority levels, where level 1 has lowest priority - level 2 the second lowest, and so on. Customers mark each packet with a priority $i \in I$. For a given time slot, let $d = (d(1), ..., d(N)$ be the aggregated allocation, where $d(i)$, $i = 1, ..., N$, is the total number of packets submitted with priority i.

The link then serves packets as follows (see Figure 1 for an illustration). All packets

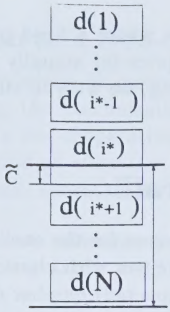


Figure 1: The link transmits all packets with a priority higher than i^*. From the packets with priority i^*, only $\tilde{C} = \left(C - \sum_{i=i^*+1}^{N} d(i)\right)$ packets are transmitted, which are chosen at random. All other packets are dropped.

with priority i such that

$$\sum_{k=i}^{N} d(k) < C$$

are transmitted. For priority i^* with

$$\sum_{i=i^*+1}^{N} d(i) < C \quad \text{and} \quad \sum_{i=i^*}^{N} d(i) \geq C,$$

only

$$\left(C - \sum_{i=i^*+1}^{N} d(i)\right)$$

packets - which are chosen at random - are transmitted. All packets with priority $i < i^*$ are dropped.

The probability $P_{tr}(i,d)$ that a packet with priority i is transmitted is then given by

$$P_{tr}(i,d) = \begin{cases} 1 & \text{if } C > \sum_{k=i}^{N} d(k) \\ \frac{C - \sum_{k=i+1}^{N} d(k)}{d(i)} & \text{if } \sum_{k=i}^{N} d(k) \geq C > \sum_{k=i+1}^{N} d(k) \\ 0 & \text{otherwise} \end{cases}$$

Let $P_{tr}(d) = (P_{tr}(1,d), ..., P_{tr}(N,d)$ the corresponding transmission probability vector.

In each time slot, customers are charged a price u_i per submitted packet with priority i; we have

$$0 < u_k < u_i, \quad \text{when } k < i, \quad i,k = 1,...,N.$$

In addition, there exists a feedback mechanism that allows customers to detect which of their packets were transmitted. Customers can then decide to retransmit dropped packets; however this will introduce additional cost. In particular, when $P_{tr}(i,d)$ is the probability that a packet with priority i is transmitted under the allocation $d = (d(1), ..., d(N))$, then the expected cost for transmitting a packet with priority i is equal to $u_i/P_{tr}(i,d)$.

The above model describes a priority service where a fixed price u_i is charged per submitted packet with priority i. However the price for actually transmitting a packet will depend on the total demand. In the following, we wish to study how customer use this service.

3 Customers with Elastic Traffic

We start out by analyzing the above priority scheme for the case where customers only care about throughput (also referred to as customers with elastic traffic), i.e. the customer's utility function depends only on the (expected) number of packets transmitted per time-slot. More precisely, when $d_r = (d_r(1), ..., d_r(N)$ is the allocation chosen by customer r, where $d_r(i)$, $i = 1, ..., N$, is the number of packets submitted with priority i, and $d_{-r} = (d_{-r}(1), ..., d_{-r}(N)$ is the aggregate allocation of all other customers, then we associate with customer r the utility U_r given by

$$U_r(d_r, P_{tr}(d_{-r} + d_r)) = \bar{U}_r(x_r),$$

where

$$x_r = \sum_{i=1}^{N} P_{tr}(i, d_{-r} + d_r) d_r(i)$$

is the expected number of packets of customer r that are transmitted and $\bar{U}_r$, is the utility of customer r under the expected throughput x_r . We assume that $\bar{U}_r$ satisfies the following assumption.

Assumption 1 *The function $\bar{U}_r$ is given by*

$$\bar{U}_r(x_r) = \int_0^{x_r} G_r(x)dx,$$

where $G_r(x) : \Re_+ \rightarrow \Re_+$ is bounded, strictly decreasing and continuously differentiable with

$$G_r(0) = u_{\max,r} > 0 \quad \textit{and} \quad \lim_{x\to\infty} G_r(x) = 0.$$

Assumption 1 implies that the function $\bar{U}_r$ is strictly concave. $G_r(x)$ is the inverse of the function $D_r : \Re_+ \rightarrow \Re_+$, where $D_r(u)$ is the optimal solution the the maximization problem

$$\max_{x\geq 0} \bar{U}_r(x) - xu.$$

$D_r(u)$ can be interpreted as the demand function associated with the utility $\bar{U}_r$, i.e. $D_r(u)$ reflects the demand of customer r in number of packets per time slot when the price per submitted packet is u and each submitted packet is transmitted with probability 1. In the following we will extensively use and refer to the function $D_r(u)$.

Given an aggregated demand $d_{-r} = (d_{-r}(1), ..., d_{-r}(N)$ by customers different from r, we will assume that customer r chooses an allocation $d_r \in \Re_+^N$ which solves the following optimization problem,

$$\max_{y_r \in \Re_+^N} \left\{ U_r(y_r, P_{tr}(d_{-r} + d_r)) - \sum_i y_r(i) u_i \right\}. \quad (1)$$

3.1 Formulation as a Non-Cooperative Game

Consider the situation where all customers are simultaneously trying to optimize their net benefit by solving the maximization problem given by Eq. (1). This situation is naturally modeled as a non-cooperative game. Here, we assume that customers use local search algorithms to find an allocation which optimizes their net benefit [1]. Accordingly, we define an equilibrium allocation for this game as follows.

Definition 1 *We call an allocation $(d_1, ..., d_R)$, $d_r \in \Re_+^N$, $r = 1, ..., R$, an equilibrium allocation when for every user $r = 1, ..., R$, d_r is a locally optimal solution to the optimization problem*

$$\max_{y_r \in \Re_+^N} \left\{ U_r(y_r, P_{tr}(d_{-r} + y_r)) - \sum_{i=1}^{N} y_r(i) u_i \right\},$$

where

$$d_{-r} = \left(\sum_{l=1}^{R} d_l \right) - d_r.$$

We can think of an equilibrium allocation as a "local" Nash equilibrium where customers seek a local rather than a global optimum. We make the following assumption to exclude the trivial case where the link capacity exceeds the the demand in the lowest priority level.

Assumption 2 *We have that*

$$\sum_{r=1}^{R} D_r(u_1) > C.$$

We have then have the following result.

Proposition 1 *Let Assumption 1 and 2 hold. Then, there exists an equilibrium allocation. Furthermore, when $(d_1, ..., d_R)$,$d_r \in \Re_+^N$, $r = 1, ..., R$, is an equilibrium allocation, then there exists a priority $i_0 \in \{1, ..., N\}$ such that*

(a) $d_r(i) = 0$, for all $r = 1, ..., R$ and all $i \notin \{i_0, i_0 + 1\}$;

(b) $\frac{u_{i_0}}{u_{i_0+1}} \leq P_{tr}(i_0, d) \leq 1$;

(c) $D_r(u_r) \leq x_r \leq D_r(u_{i_0})$, for all $r = 1, ..., R$, where

$$u_r = \frac{u_{i_0}}{P_{tr}(i_0, d)} \left(1 - \frac{d_r(i_0)}{d(i_0)} \right)^{-1};$$

(d) $\sum_r x_r = C$;

where

$$d = \sum_{r=1}^{R} d_r \quad \text{and} \quad x_r = \sum_{i=1}^{N} d_r(i) P_{tr}(i, d),$$

and where we use the convention that $u_{N+1} = \max_{r=1,...,R} u_{\max,r}$, for $u_{\max,r}$ as given in Assumption 1.

---

[1] To do this, customers may probe the link to discover how to change their allocation in order to improve their net benefit

Proposition 1 states that there always exists an equilibrium allocation, which, however, is not necessarily unique. Let us briefly comment on the Properties (a)-(d) of Proposition 1,

(a) states that, under every equilibrium allocation, at most two different priorities are being used.

(b) states that when priority i is used under a given equilibrium allocation, then the probability that a submitted packet with priority i gets transmitted is lower bounded by u_i/u_{i+1}. This suggests that pricing can be used to control link congestion (see discussion in Section 5).

(c) implies that under a given equilibrium allocation, the lower bound on the expected transmission rate x_r is equal to the demand $D_r(u_r)$ for the price

$$u_r = \frac{u_{i_0}}{P_{tr}(i_0, d)}\left(1 - \frac{d_r(i_0)}{d(i_0)}\right)^{-1}.$$

Therefore, when for customer r the ratio $d_r(i_0)/d(i_0)$ is negligible small (what we will call a "small customer"), then the price u_r is (approximately) equal to $u_{i_0}/P_{tr}(i_0, d)$. Furthermore, using (b), we obtain that in this case we have

$$D_r(u_{i_0+1}) \leq x_r \leq D_r(u_{i_0}).$$

One interpretation of this result is that the priority scheme considered here protects small customers, i.e. under an equilibrium allocation, small customers are guaranteed to obtain an expected rate $x_r \in [D_r(u_{i_0+1}), D_r(u_{i_0})]$.

(d) states that under every equilibrium allocation the whole link capacity gets used.

3.2 Competitive Price-Taking Assumption

In the previous subsection we assumed that, while choosing an allocation, customers anticipate how they affect the transmission probabilities (see Eq. (1)). When there is no analytical model available, this can be done by probing the link to obtain estimates of the gradient of $P_{tr}\left(\sum_r d_r\right)$ with respect to d_r, where $d = (d_1, ..., d_R)$ is the current aggregate allocation over all customers. Unfortunately, such gradient estimates tend to have a large variance which makes this procedure computationally expensive.

As an alternative approach, customers may ignore how they influence the transmission probabilities and tune their allocation as follows;

1. For the current allocation $(d_1, ..., d_R)$, probe the network to obtain (estimates of) the transmission probabilities $P_{tr}(\sum_r d_r) = (P_{tr}(1, \sum_r d_r), ..., P_{tr}(N, \sum_r d_r))$.
2. Update the current allocation $d_r \in \Re^N_+$ towards an optimal solution of optimization problem

$$\max_{y_r \in \Re^N_+} \left\{ U_r\left(y_r, P_{tr}\left(\sum_r d_r\right)\right) - \sum_{i=1}^{N} y_r(i) u_i \right\}. \tag{2}$$

In general, it is easier to estimate the transmission probability $P_{tr}(i, \sum_r d_r))$ than to estimate the gradient of $P_{tr}(i, \sum_r d_r)$ with respect to d_r. An optimal solution to Eq. (2) can easily be computed, i.e. an optimal solution is obtained by choosing

$$d_r(i) = \begin{cases} D_r\left(u_{i^*}/P_{tr}(i^*, d)\right) & \text{if } i = i^*, \\ 0 & \text{otherwise} \end{cases}$$

where the priority i^* is such that

$$u_{i^*}/P_{tr}(i^*,d) \leq u_i/P_{tr}(i,d), \quad \text{for all } i = 1, ...N.$$

The simplifying assumption that customers ignore how they influence the transition probabilities corresponds to the standard competitive price-taking assumption of economic theory. In this context, it can be justified when

(a) many "small" customers are sharing the link capacity so that the influence of an individual customer on the transmission probabilities is negligible.

(b) it is impossible (or too expensive) for a customer to determine how the choice of an allocation may affect the transmission probabilities.

We then define an equilibrium allocation as follows.

Definition 2 *We call an allocation* $(d_1, ..., d_R)$, $d_r \in \Re^N_+$, $r = 1, ..., R$, *an equilibrium allocation under the competitive price-taking assumption when for every user* $r = 1, ..., R$, d_r *is an optimal solution to the optimization problem*

$$\max_{y_r \in \Re^N_+} \left\{ U_r \left(y_r, P_{tr} \left(\sum_{r=1}^{R} d_r \right) \right) - \sum_{i=1}^{N} y_r(i) u_i \right\}.$$

We have the following result.

Proposition 2 *Let Assumption 1 and 2 hold. Then there exists an equilibrium allocation under the competitive price-taking assumption. Furthermore, when* $(d_1, ..., d_R)$,$d_r \in \Re^N_+$, $r = 1, ..., R$, *is an equilibrium allocation under the competitive price-taking assumption, then there exists a priority* $i_0 \in \{1, ..., N\}$ *such that*

(a) $d_r(i) = 0$, *for all* $r = 1, ..., R$ *and all* $i \notin \{i_0, i_0 + 1\}$;

(b) $\frac{u_{i_0}}{u_{i_0+1}} \leq P_{tr}(i_0, d) \leq 1$;

(c) $x_r = D_r(u^*)$, *for all* $r = 1, ..., R$, *where* $u^* = \frac{u_{i_0}}{P_{tr}(i_0,d)}$;

(d) $\sum_r x_r = C$;

where

$$d = \sum_{r=1}^{R} d_r \quad \textit{and} \quad x_r = \sum_{i=1}^{N} d_r(i) P_{tr}(i, d),$$

and where we use the convention that $u_{N+1} = \max_{r=1,...,R} u_{\max,r}$.

This result is Proposition 1 specialized to the case where

$$\frac{d_r(i_0)}{d(i_0)} = 0, \quad \text{for all } r = 1, ..., R.$$

The price u^* in Proposition 2 can be interpreted as a "congestion price", i.e. the expected rate x_r of each customer r is equal to the demand under the congestion price u^*. The scheme is then "fair" in the sense that all customers see the same congestion price u^*.

4 Customers with QoS-Sensitive Traffic

In this section, we extend our analysis to allow customers with QoS-sensitive traffic. We model this situation by adding to the customer's utility function a penalty term which is sensitive towards link congestion.

Consider a fixed customer r and let $\bar{U}_r(x)$ be an utility function which satisfies Assumption 1. In addition, let $d_r = (d_r(1), ..., d_r(N)$ be the allocation of customer r and let $d_{-r} = (d_{-r}(1), ..., d_{-r}(N))$ be the aggregate allocation by customers different than r. We then associated with customer r the utility $U_r(d_r, P_{tr}(d_{-r} + d_r)$ given by

$$U_r\Big(d_r, P_{tr}(d_{-r} + d_r)\Big) = \bar{U}_r(x_r) - \sum_{i=1}^{N} c_r\Big(P_{tr}(i, d_{-r} + d_r)\Big)d_r(i), \tag{3}$$

where

$$x_r = \sum_{r=1}^{R} P_{tr}(i, d_{-r} + d_r)d_r(i)$$

and the function $c_r : [0, \infty) \rightarrow [0, \infty)$ is non-increasing and continuous, with

$$c_r(\rho) = 0, \quad \text{for } \rho \geq 1.$$

The function $c_r\Big(P_{tr}(i, d + d_r)$ models the QoS requirements of user r. When c_r is equal to 0 for all $\rho \geq 0$, we obtain the utility function of the previous section where customers only cared about throughput. A customer with with tight QoS requirements can be modeled by letting the function $c_r(\rho)$ increase sharply when ρ falls below 1.

We study this situation under the competitive price-taking assumption, i.e. we wish to identify an allocation $(d_1, ..., d_R)$ such that for every user $r = 1, ..., R$, d_r is an optimal solution to the optimization problem

$$\max_{y_r \in \Re^N_+} \left\{ U_r\left(y_r, P_{tr}\left(\sum_{r=1}^{R} d_r\right)\right) - \sum_{i=1}^{N} y_r(i)u_i \right\},$$

where the utility function U_r is given by Eq. (3). We have the following result.

Proposition 3 *Let Assumption 1 and 2 hold. Then there exists an equilibrium allocation under the competitive price-taking assumption. Furthermore, when $(d_1, ..., d_R)$,$d_r \in \Re^N_+$, $r = 1, ..., R$, is an equilibrium allocation under the competitive price-taking assumption, then there exists a priority $i_0 \in \{1, ..., N\}$ such that*

(a) $d_r(i) = 0$, for all $r = 1, ..., R$ and all $i \notin \{i_0, i_0 + 1\}$;

(b) $\frac{u_{i_0}}{u_{i_0+1}} \leq P_{tr}(i_0, d) \leq 1$;

(c) $D_r(u_r) \leq x_r \leq D_r(u_{i_0})$, for all $r = 1, ..., R$, where

$$u_r = \min\left\{ u_{i_0+1}, \frac{u_{i_0} + c_r\Big(P_{tr}(i_0, d)\Big)}{P_{tr}(i_0, d)} \right\};$$

(d) $\sum_r x_r = C$;

where

$$d = \sum_{r=1}^{R} d_r \quad \text{and} \quad x_r = \sum_i d_r(i) P_{tr}(i,d),$$

and where we use the convention that $u_{N+1} = \max_{r=1,...,R} u_{\max,r}$.

This result is the equivalent of Proposition 2, however the price u_r, which determines the lower bound on the expected rate x_r, now depends on customer's QoS requirements, i.e. the penalty term $c_r\Big(P_{tr}(i_0,d)\Big)$. Customers with tight QoS requirement are willing to pay a higher price to obtain a better QoS service and submit their packets with priority i_0+1. Customer with elastic traffic can tolerate packet loss and will submit their packets with priority i_0. In this sense, the priority scheme considered here enables the provision of differentiated QoS.

5 Pricing as a Tool to Control Link Congestion

Propositions 1- 3 suggest that pricing can be used to control link congestion. In particular, when for the priority i_0 of Propositions 1- 3 we have that $i_0 < N$, then the lower bound on the probability that a submitted packets gets transmitted under an equilibrium allocation is purely a function of the prices $u_1, ..., u_N$. To ensure that for every equilibrium allocation we have $i_0 < N$, by Property (d) of Propositions 1- 3, it suffices to choose the price u_N high enough so that

$$\sum_r D_r(u_N) < C.$$

This suggest to following procedure to guarantee that under every equilibrium allocation a submitted packets get transmitted with a probability greater or equal to P_0,

1. choose the price u_N so that $\sum_r D_r(u_N) < C$, and,
2. for $i = 1, ..., N$, set

$$u_i = u_N P_0^{N-i}.$$

Generally, the number of customers accessing the link and their utility function will change over time. Let $R(t)$ is the number of customers accessing the link at time $t \geq 0$ and $D_r(u,t)$, $r = 1, ..., R(t)$, are the corresponding demand functions. To ensure that the probability that a submitted packets gets transmitted never falls below P_0 under an equilibrium allocation, the prices $u_1, ..., u_N$ have to be chosen such that the following holds,

$$\max_{i \in \{1,...,N-1\}} u_i/u_{i+1} \geq P_0 \quad \text{and} \quad \sum_r D_r(u_N, t) < C, \quad \text{for all } t \geq 0.$$

As it is difficult to predict the aggregated link demand (especially over a long time horizon), one would expect that either the prices associated with the different priorities, or the link capacity C, have to be updated from time to time to track the conditions above. However, by starting with a conservative choice of u_N, i.e. by choosing u_N such that $\sum_r D_r(u_N) << C$, these updates may have to be done only infrequently (for example every month) - making the implementation of this priority scheme simpler than the one considered in [3] where prices associated with the different priorities levels are dynamically updated on a fast time-scale (in the order of seconds).

6 Future Work

Regarding future work, it would be interesting to investigate whether the existence of multiple equilibria can lead to oscillations. Related to that issues is the question whether there exists an algorithm which ensures that the customer's allocations converge to an equilibrium.

References

[1] S. Chen and K. Park, *"An Architecture for Noncooperative QoS Provision in Many-Switch Systems,"* Proc. IEEE INFOCOM '99, 1999.

[2] R. Cocchi, S. Shenker, D. Estrin, and L. Zhang, *"Pricing in Computer Networks: Motivation, Formulation, and Example,"* IEEE/ACM Transactions on Networking, 1, 1993.

[3] A. Gupta, D. O. Stahl, and A. B. Whinston, *"A Stochastic Equilibrium Model of Internet Pricing,"* Journal of Economics Dynamics and Control, 21:697–722, 1997.

[4] J. Heinanen, F. Baker, W. Weiss, and J. Wroclawski, *" Assured forwarding phb group,"* IETF: RFC 2597, ftp://ftp.isi.edu/in-notes/rfc2597.txt, 1999.

[5] A. Odlyzko, *"Paris Metro Pricing for the Internet"*, http://www.research.att.com/~amo, 1998.

[6] K. Park, M. Sitharam, and S. Chen, *"Quality of Service Provision in Noncooperative Networks: Heterogenous Preferences, Multi-Dimensional QoS Vectors, and Burstiness,"* Proc. International Conference on Information and Computation Economies, 1998.

[7] Ch. Semeria and F. Fuller, *"3com's Strategy for Delivering Differentiated Service Levels,"* http://www.3com.com/technology/tech_net/white_papers/500652.html, 1999.

Comparing tandem queueing systems and their fluid limits

Eitan ALTMAN[†] Ger KOOLE[‡] Tania JIMÉNEZ[†§]

† INRIA,2004 route des Lucioles, B.P. 93, 06902 Sophia Antipolis, France
‡ Vrije Universiteit, De Boelelaan 1081a, 1081 HV Amsterdam, The Netherlands
§ CeSiMo, Universidad de Los Andes, Mérida, Venezuela

Abstract

In this paper we compare queueing systems and their fluid limits. It is shown that the expected workload (and certain functions of the workload) is higher in the queueing system than in the fluid approximation. The queueing systems that we consider consist of tandems of multi-server queues with general service times and Markov-modulated arrivals. The analysis is based on dynamic programming and the use of phase-type distributions. Numerical results to illustrate the results are given.

1 Introduction

Queueing theory is mainly concerned with calculating the influence of randomness in certain processes as a function of certain stationary input processes, that model arrivals and departures. Indeed, the variance in for example service times has a great influence on waiting times, as is nicely illustrated for the M/G/1 queue in the celebrated Pollaczek-Khitchine formula.

However, in certain situations we do not wish to make the stationarity assumption, but to study the influence of changes in for example arrival rate. In such cases exact queueing results, taking into account non-stationarity and variances, are extremely hard to obtain. An "engineering approach" [3, p. 56] to this problem is to focus on the non-stationarity, and assuming continuous deterministic arrival and departure flows. This leads to the well-known fluid approximation for queues.

After having analyzed the fluid approximation, the question remains to what extent this analysis is valid for the original model. In this paper we try to answer part of this question by showing that for certain systems and performance measures the fluid limit performs better than the original queueing system. We show this for tandems of multi-server queues and performance measures such as total (weighted) workload. Service times are general but fixed for each customer. There is an environment modulating the arrival rates and the server speeds.

In Section 2 our main result is formulated in detail. This result is obtained in Sections 3 through 5, as follows. First we study exponential queueing models. This queueing model, introduced in Section 3, consists of a tandem of multi-server queues. Both the arrival rate and the service rates are governed by an underlying Markov process. Arrivals can occur in batches.

programming is used to compare two tandem queueing models that differ in the workload: one system has batch size one, the other possibly higher batch sizes but less frequent arrivals. We take the arrival rates and the number of exponential phases per customer such that the average workload to the systems is equal. We give conditions on the performance measures for which the system with batch arrivals behaves worse. In Section 5 the fluid limit is studied. By increasing service rates, arrival rates and batch sizes in the correct way the queueing model without batches converges to a fluid model, and the model with batches converges to a workload model with arbitrary service times. This comparison is illustrated with the analysis of a few example systems in Section 6. One is a single queue model with non-stationary arrival rate, motivated by [9]. Another is the comparison of standard homogeneous M/M/1, M/D/1 queues and their fluid approximations with a large initial workload. The last consists of a tandem system with two queues with a large amount of initial workload in the first one.

The full length version of this paper will also contain a section in which similar results are obtained for event graphs, using theory on (max,+) algebras and Petri nets.

2 Performance measures and main result

In this section we state the main result. For certain tandem systems we give conditions on the performance measures under which the fluid limit has a performance lower than the original queueing system. These conditions are formulated in terms of sets of functions. We first discuss these sets. For the moment, all we need to know about our system is that it consists of m queues and an environment.

All performance measures (or costs, in dynamic programming terminology) are functions of the whole state space, which is given by $\mathbb{N}_0 \times \mathbb{R}_+^m$. The 0th (integer) state component represents the environment, the ith state component ($1 \leq i \leq m$) the workload in the ith queue. Each performance measure is an element of the set $\mathcal{F} = \{f | f : \mathbb{N}_0 \times \mathbb{R}_+^m \to \mathbb{R}\}$.

Definition 2.1 *A function* $f : \mathbb{R}_+^m \to \mathbb{R}$ *is called* directionally convex *if for all* $x, y, z \in \mathbb{R}_+^m$

$$f(x+y) + f(x+z) \leq f(x) + f(x+y+z).$$

Directional convexity was introduced in [8], where it is formulated in a slightly different (but equivalent) way.

To keep terminology simple, we call a function $f \in \mathcal{F}$ also directionally convex (written as $f \in \mathcal{F}(dc)$) if f is directionally convex for each environment state. This assures that if $f \in \mathcal{F}(dc)$, then there is no condition on the way f depends on the environment.

Many performance measures are directionally convex. Note that each performance measure that is additive and convex in the components is also directionally convex.

Definition 2.2 *A function* $f : \mathbb{R}_+^m \to \mathbb{R}$ *is called* downstream decreasing *if for all* $1 \leq i < j \leq m$,

$$f(x+e_j) \leq f(x+e_i)$$

and if

$$f(x) \leq f(x+e_m).$$

f is downstream decreasing for a fixed environment state. It is easy to show that every downstream decreasing function is increasing (terms such as increasing and decreasing are used in the non-strict sense) in each of the variables. We write $\mathcal{F}(dc, dd) = \mathcal{F}(dc) \cap \mathcal{F}(dd)$.

Costs that are downstream decreasing imply that it is favorable to have jobs further down the line. This is the case in many systems. Some examples where it is not the case is in systems where holding costs increase down the line, as in some production systems, and in systems with finite buffers, modeled by additional costs if queue lengths exceed buffer levels.

We describe in words the model. We consider a tandem system consisting of $m \in \mathbb{N}$ multi-server queues and arrivals at one or more queues. Customers have general service times (with average 1), that remain equal at the different stages of the tandem system. The speeds at which the servers at different queues work can differ. There is an independent Markovian environment governing arrival rates and server speeds. The fluid model consists also of m queues and the same Markovian environment. The arrival rates are now the speeds at which fluid enters the system.

Note that the condition that the average service time is 1 does not constrain the model: by rescaling we can get any average. The constraint is that the queueing and the fluid model have the same expected offered workload.

Now follows our main result. It will be proven in the next three sections. For an initial workload w, let W_t be the workload vector of the queueing system at time t. Similarly, let Z_t be the workload of the fluid model at time t, also for the initial state w. With U_t we indicate the state of the environment at t, and $U_0 = u$.

Theorem 2.3 *If $C \in \mathcal{F}(dc, dd)$ and if $\inf_u C(u, 0) > -\infty$, then*

$$\mathbb{E}_{u,w} C(U_t, Z_t) \leq \mathbb{E}_{u,w} C(U_t, W_t).$$

3 The exponential model

To derive Theorem 2.3, we study first exponential queueing models with possibly batch arrivals. In this section we introduce these systems in detail. They consist of $m \in \mathbb{N}$ queues and an environment. The environment is modeled by an arbitrary continuous-time Markov process on $\mathbb{N}_0$, with transition rates $\gamma(u, v)$, with $\sum_{v=0}^{\infty} \gamma(u, v)$ uniformly bounded. This Markov process governs the transitions at the queueing system. At each of the m queues arrivals can occur, at a rate $\lambda^j(u)$ at queue j, with u the current state of the environment. We assume that $\sup_u \lambda^j(u) < \infty$ for each j. Queue j, $1 \leq j \leq m$, has S_j servers, that serve at a rate $\mu^j(u)$ each. Again, $\sup_u \mu^j(u) < \infty$. This specifies the model without batch arrivals. The model with batch arrivals will be introduced later in this section.

Let $X_t \in \mathbb{N}_0^m$ denote the number of jobs in the queues at t. Next we develop a dynamic programming formulation for $\mathbb{E}_{u,x} C(U_t, X_t)$, the expected costs at t for the initial state of the system $(u, x) \in \mathbb{N}_0^{m+1}$. Note that $C : \mathbb{N}_0^{m+1} \to \mathbb{R}$ in this section and in the next. Because all transition rates are uniformly bounded we can rewrite them, for some $\alpha > 0$, as $\gamma(u, v) = \alpha \bar{\gamma}(u, v)$, $\lambda^j(u) = \alpha \bar{\lambda}^j(u)$, and $\mu^j(u) = \alpha \bar{\mu}^j(u)$, with $\sum_{v=0}^{\infty} \bar{\gamma}(u, v) + \sum_{j=1}^{m} (\bar{\lambda}^j(u) + S_j \bar{\mu}^j(u)) \leq 1$. By adding dummy transition to the environment, i.e., giving $\bar{\gamma}(u, u)$ a positive value, we can assure that $\sum_{v=0}^{\infty} \bar{\gamma}(u, v) + \sum_{j=1}^{m} [\bar{\lambda}^j(u) + S_j \bar{\mu}^j(u)] = 1$ for every u.

at rate α jumps are generated, and at the jump times the system changes state according to a discrete-time Markov chain with transition probabilities $\bar{\gamma}(u, v)$, $\bar{\lambda}^j(u)$, and $\bar{\mu}^j(u)$, where one or more "dummy departures" occur at queue j if the state is such that there are less than S_j jobs in queue j. This representation of the system will help us find a dynamic programming formulation for $\mathbb{E}_{u,x} C(U_t, X_t)$, the expected costs at t.

But first we introduce a condition that assures that $\mathbb{E}_{u,x} C(U_t, X_t)$ is integrable for all $C \in \mathcal{F}(dd)$.

Lemma 3.1 *For all $C \in \mathcal{F}(dd)$ that satisfy $\inf_{u \in \mathbb{N}_0} C(u, 0) > -\infty$, $\mathbb{E}_{u,x} C(U_t, X_t)$ exists and is finite or ∞.*

Proof Every $C \in \mathcal{F}(dd)$ is increasing in all its components, therefore $C(u, 0) \leq C(u, x)$ for all x and u. Thus $\inf_u C(u, 0) = \inf_{u,x} C(u, x)$, and $\mathbb{E}_{u,x} C(U_t, X_t) \geq \inf_{u \in \mathbb{N}_0} C(u, 0)$. □

By conditioning on the number of jumps of the Poisson process in $[0, t]$, we can write $\mathbb{E}_{u,x} C(U_t, X_t)$ as

$$\mathbb{E}_{u,x} C(U_t, X_t) = \sum_{n=0}^{\infty} \frac{(\alpha t)^n}{n!} \mathrm{e}^{-\alpha t} V_n(u, x), \tag{1}$$

with $V_n(u, x)$ the expected costs after n steps of the discrete-time model. For $V_n(u, x)$ we will now give a dynamic programming recursion. (The same method is also used, for different applications, in [4] (Section 5.2) and [7] (Section 2.3).)

For $n = 0$ we have to take $V_0(u, x) = C(u, x)$, with C the performance measure or costs. For $n > 0$ the value function V_n can be recursively defined by:

$$V_{n+1}(u, x) =$$

$$T_{env}\Big(V_n, T_{A(1)}(V_n), \ldots, T_{A(m)}(V_n), T_{TS(1)}(V_n), \ldots, T_{TS(m-1)}(V_n), T_{D(m)}(V_n)\Big)(u, x),$$

with

- $T_{A(i)} f(u, x) = f(u, x + e_i)$;
- $T_{D(i)} f(u, x) = \begin{cases} x_i f(u, x - e_i) + (S_i - x_i) f(u, x) & \text{if } x_i < S_i, \\ S_i f(u, x - e_i) & \text{otherwise;} \end{cases}$
- $T_{TS(i)} f(u, x) = \begin{cases} x_i f(u, x - e_i + e_{i+1}) + (S_i - x_i) f(u, x) & \text{if } x_i < S_i, \\ S_i f(u, x - e_i + e_{i+1}) & \text{otherwise;} \end{cases}$
- $T_{env}(f_0, \ldots, f_{2m})(u, x) = \sum_{v \in \mathbb{N}_0} \bar{\gamma}(u, v) f_0(v, x) + \sum_{j=1}^{m} [\bar{\lambda}^j(u) f_j(u, x) + \bar{\mu}^j(u) f_{m+j}(u, x)]$.

Let us interpret this definition of V_{n+1}. All Ts stand for possible events in the system. T_{env}, the environment operator, specifies the type of event that occurs at this jump. If it is a change of state of the environment, then the state changes from (u, x) to (v, x) and there are n more jumps to go, thus expected costs $V_n(v, x)$. This is reflected by the fact that by inserting the definition of T_{env} in V_{n+1} we find $f_0 = V_n$. If on the other hand an arrival occurs, say at queue j, then the next event $T_{A(j)}$ occurs, which event is applied to V_n. The same holds for $T_{TS(i)}$, the event operator corresponding to S servers moving customers from queue i to queue $i + 1$ (TS stands for "tandem server"), and $T_{D(m)}$, the event operator modeling the departures from queue m. Note that the coefficients sum to 1, because the coefficients in the departure operators sum to the number of servers.

Splitting up the dynamic programming equation into smaller event operators as we just did simplifies the proofs in the next section. For full details of this method we refer to [5]; here we use it in a simplified form, adapted to the comparison of different systems.

differs from the system with value function V_n in the arrivals at queue i. Under V_n^B arrivals occur less frequently, but in batches. To be more specific, $T_{A(i)}$ of V_n is replaced by

- $T_{BA(i)}f(u,x) = \sum_{k=0}^{\infty} p_k^i f(u, x + ke_i)$.

Here we take the $p_k^i \geq 0$ such that $\sum_{k=0}^{\infty} p_k^i = 1$. Define $\bar{p}^i = \sum_{k=0}^{\infty} kp_k^i$.

In the next section we compare V_n and V_n^B. But let us try to understand why $T_{BA(i)}$ is an interesting operator to consider. It is known ([1, 11]) that any probability distribution can be approximated arbitrarily close by a mixture of Erlang or gamma distributions, all with the same parameter. With $T_{BA(i)}$ we can model this approximation. When we increase the accuracy of the approximation, the batch sizes increase, but the service time per exponential unit decreases. In the model without batch arrivals we keep the expected number of exponential units (almost) equal, meaning an increasing arrival rate. Of course the service time per exponential unit decreases, and thus the total expected offered workload remains the same, and converges to a fluid process, as we will see in Section 5. We show in the next section that if $C \in \mathcal{F}(dc, dd)$ and $\bar{p}^i \geq 1$ for all i, then $V_n(u,x) \leq V_n^B(u,x)$ for all n, u and x. From this the comparison at t follows.

4 Comparison

Let X_t^B denote the state of the system with batch arrivals at time t. The main result of this section is the following.

Theorem 4.1 *If* $C \in \mathcal{F}(dc, dd)$, $\bar{p}^i \geq 1$ *for all* i, *and* $\inf_{u\in\mathbb{N}_0} C(u,0) > -\infty$, *then*

$$\mathbb{E}_{u,x} C(U_t, X_t) \leq \mathbb{E}_{u,x} C(U_t, X_t^B).$$

Proof Below, in Theorem 4.5, we will show that $V_n(u,x) \leq V_n^B(u,x)$ if $C \in \mathcal{F}(dc, dd)$ for all n. Then the result follows immediately by using (1) and observing that this equation also holds for $\mathbb{E}_{u,x} C(U_t, X_t^B)$. □

In the proof of Theorem 4.5 we make use of certain properties of directionally convex functions and of the fact that V_n itself is directionally convex. The latter is shown for a more general model in [6], Corollary 4.6.

We will use the following useful characterization of directionally convexity on $\mathbb{N}_0^m$ (e.g., see [6]). We call a function f componentwise convex in component i if

$$2f(x+e_i) \leq f(x) + f(x+2e_i)$$

for all x, and supermodular in i and j if

$$f(x+e_i) + f(x+e_j) \leq f(x) + f(x+e_i+e_j)$$

for all x. Lemma 3.2 of [6] states that a function is directionally convex if and only if it is componentwise convex and supermodular in all components i with $1 \leq i, j \leq m$. (Note that componentwise convexity is in fact supermodularity with $i = j$.)

Furthermore, in Lemma 3.8 of [6] it is shown that every downstream decreasing function is also increasing in each component, i.e., $f(x) \leq f(x+e_i)$ for all x and i. The following lemma is a special case of Corollary 4.6 of [6].

The difference between the last part of Corollary 4.6 in [6] and the current theorem is that the environment in Corollary 4.6 is more general, also a Markov arrival process can be modeled with it. This is not useful in the context of fluid queues.

Now we show certain properties of componentwise convexity, as defined already in Section 2. We denote componentwise convexity in i with $CC(i)$.

For a function $f : \mathbb{N}_0^m \to \mathbb{R}$, we define $\overrightarrow{f}^i : \mathbb{R}_+^m \to \mathbb{R}$ to be the function which coincides with f on $\mathbb{N}_0^m$, and is the linear interpolation of f along the direction i. If f is $CC(i)$ then the derivatives of $\overrightarrow{f}^i$ in direction i are clearly monotone increasing, so that

$$f \text{ is CC(i) } \Rightarrow \overrightarrow{f}^i \text{ is convex in direction } i \tag{2}$$

This implies the following:

Lemma 4.3 *The function f is $CC(i)$ if and only if*

$$f(x + \alpha k e_i) \leq (1 - \alpha) f(x) + \alpha f(x + k e_i),$$

for all $x \in \mathbb{N}^m$ and $k \in \mathbb{N}$ and $\alpha \in [0,1]$ such that $\alpha k \in \mathbb{N}$.

Proof The "if"-part follows directly by taking $k = 2$ and $\alpha = 1/2$. The "only if"-part follows from (2). □

The inequality proven in the next lemma will prove to be crucial in the proof of Theorem 4.5.

Lemma 4.4 *If f is $CC(i)$ and increasing in i, $\sum_{k=0}^{\infty} p_k = 1$, and $\sum_{k=0}^{\infty} k p_k \geq 1$, then*

$$f(x + e_i) \leq \sum_{k=1}^{\infty} p_k f(x + k e_i).$$

Proof Since $\sum_{k=0}^{\infty} k p_k \geq 1$ it follows directly from Jensen's inequality. □

Theorem 4.5 *If $C \in \mathcal{F}(dc, dd)$ and $\bar{p}^i \geq 1$, then $V_n(u, x) \leq V_n^B(u, x)$ for all $n \in \mathbb{N}_0$.*

Proof We prove the theorem by induction to n. For $n = 0$ we have $V_0(u, x) = V_0^B(u, x) = C(u, x)$. Assume that it is proven up to $n \geq 0$. Consider V_{n+1}. From $V_n(u, x) \leq V_n^B(u, x)$ it follows directly that $T_{A(i)} V_n(u, x) \leq T_{A(i)} V_n^B(u, x)$, $T_{TS(i)} V_n(u, x) \leq T_{TS(i)} V_n^B(u, x)$, and $T_{D(m)} V_n(u, x) \leq T_{D(m)} V_n^B(u, x)$. The crucial step is showing $T_{A(i)} V_n(u, x) \leq T_{BA(i)} V_n^B(u, x)$. As $C \in \mathcal{F}(dc, dd)$, it follows from Lemma 4.2 and Lemma 4.4 that

$$V_n(u, x + e_i) \leq \sum_{k=0}^{\infty} p_k V_n(u, x + k e_i).$$

By induction $V_n(u, x) \leq V_n^B(u, x)$ for all (u, x), and thus holds

$$V_n(u, x + e_i) \leq \sum_{k=0}^{\infty} p_k V_n^B(u, x + k e_i).$$

This is exactly $T_{A(i)} V_n(u, x) \leq T_{BA(i)} V_n^B(u, x)$. As T_{env} simply consists of a convex combination of terms for which the inequality holds, $V_{n+1}(u, x) \leq V_{n+1}^B(u, x)$ easily follows. □

We worked out the comparison for a single operator. Repeating it for multiple arrival operators is straightforward.

There are different ways to construct and interpret fluid limits. We prefer to see them as a means to study transient behavior, such as peak hours. This is the classical interpretation, see e.g. [9, 3]. The non-stationarity can be modeled effectively with the environment. For this reason a construction involving scaling time (such as in [2]) is not appropriate, as it would mean scaling the environment as well. Instead, we see the fluid limit as a simultaneous increase of the arrival rate and a decrease of the size of the workload per job. This corresponds to the average over an infinite number of realizations of the arrival process, for each realization of the environment separately.

We construct a series of comparisons between two queueing systems that is such that one of the better system converges to the fluid limit and the worse system to the workload vector. The comparison holds at each stage by Theorem 4.1.

Before going into the details of the comparison, we define what we call a fluid process. We say that the offered workload O_t is a fluid process generated by the environment U_t if there is a rate function $\lambda : \mathbb{N}_0 \to \mathbb{R}_+$ such that $O_t = \int_0^t \lambda(U_s)ds$. Thus a fluid is characterized by the absence of jumps and, for a given realization of the environment process, a deterministic growth.

Now add an index k to the environment, giving a series $\{U_t^k\}_{k=1}^{\infty}$, where U_t^k is equal to U_t except that all $\lambda^1(u)$ and $\mu^j(u)$ are multiplied by a factor k. Consider the queueing system without batch arrivals. We multiplied the server speeds with k. An alternative of looking at this system is assuming that the server speeds remained unchanged, but that each exponential task has a service time divided by k, i.e., with rate k. Thus the offered workload generated by U_t^k up to t consists of a number of exponential phases with rate k, where this number is Poisson distributed with mean $\int_0^t k\lambda^1(U_s)ds$. Then, from standard properties of the Erlang distribution, the offered workload converges, as $k \to \infty$, to a fluid with rate $\lambda^1(U_t)$.

Next we consider the system with batch arrivals. We want to have the same batch arrival process, independent of the environment. This is achieved by taking $p_0^1 = 1 - 1/k$ for the kth approximation. Thus on average $T_{BA(1)}$ selects 1 out of k points from the process with rate $k\lambda^1(U_t)$. This results in an arrival process with rate $\lambda^1(U_t)$. The service time distribution, with distribution function G, is approximated by a series of mixtures of Erlang distributions, with distribution function G^k of the kth approximate defined by

$$G^k(x) = \sum_{n=1}^{\infty} \beta_n^k E_n^k(x),$$

where $\beta_n^k = G(n/k) - G((n-1)/k)$, and $E_n^k(x)$ the distribution function of the Erlang distribution with rate k and n phases. The decrease in service time of the exponential phases is already incorporated in the server speeds. The operator $T_{BA(1)}$ generates the right number of phases if we take $p_n^1 = \beta_n^k/k$ for all $n > 0$. It is important to note that $\bar{p}^1 = \sum_{n=0}^{\infty} np_n^1 = \sum_{n=0}^{\infty} n/k(G(n/k) - G((n-1)/k)) \geq \int_0^{\infty} dG(x)$. As the average service time equals 1, we find $\bar{p}^1 \geq 1$. With other words, for suitable C, we are allowed to apply Theorem 4.1 for every U_t^k and $T_{BA(1)}$ as defined above. This means that the comparison holds for the limit as well. We have already shown that the queueing model without batch arrivals converges to the fluid model. Thus it remains to show that the batch arrival model converges to the original queueing model. As the workload at t is a continuous function of the input, it suffices to show that G^k converges to G. This result if we have finished proving Theorem 2.3; there is one complication left however.

In X_t^B the tasks of which a batch consists are treated separately, while they should be treated as a whole. In the current systems one task of a batch can be processed at one stage while another task is processed at the next stage already. The same holds for multi-server queues: tasks of the same customer can be processed in parallel. Thus the correct system is yet another that delays certain tasks until all tasks of a customer have finished processing.

The solution is as follows. We introduce yet another system that incorporates the delays. To show that the real system is even worse than the one we already studied with T_{BA} we have to show that it is better to have tasks further down the line. For performance measures in $\mathcal{F}(dd)$ it can easily be shown using a coupling argument that the delayed system has a worse performance. This completes the proof of Theorem 2.3.

6 Numerical examples

In the introduction we already stated that in queueing we can distinguish two types of variations: changes in the system parameters such as arrival and service rates, and the random changes in interarrival and service times. The first type of variation is modeled by the environment, and therefore it already shows up in the fluid approximation. The queueing system also captures the second type of variations.

In this section we compare queueing systems and their fluid approximations for several configurations. First note that a standard stable M/G/1 queue has fluid approximation 0 for an initially empty system. If there is some initial workload then the system will reach and stay at 0 after some point in time. This shows that it is only interesting to consider fluid approximations if there is a large initial workload or if there are time intervals at which the offered workload is higher than the service capacity in one or more of the queues. In the configurations that we consider one of the two situations will always be the case. In all cases we take a deterministic environment. Of course, this cannot be modeled immediately by the Markovian environment that we studied in previous sections. However, using arguments comparable to those in [1], we see that any deterministic environment can be approximated arbitrarily close by a Markovian environment.

The first example is in the same spirit as that of [9, Figure 2.2] and [3, Figure 2.7]. It consists of a single M/M/1 queue with a constant service rate of 0.4, and arrival rate 0.8 up to time 100 and arrival rate 0.2 after 100. The initial workload is equal to 0. For this system it is trivial to calculate the fluid approximation. Using simulation we calculated the expected workload at each time up to 600. The results can be found in Figure 1. The queueing system is the average over 600 traces. We see that the average workload in the queueing system is always higher than the workload in the fluid system, as predicted by Theorem 2.3. We also see that the behavior is similar for large queue lengths. We also see that the fluid is equal to 0 from 300 on, while the expected workload in the M/M/1 system converges to 2.5, the stationary workload of the M/M/1 queue with rates 0.2 and 0.4. The fluid approximation is independent of the service time distribution, it depends only on the expectation of the service time. However, the workload in the M/G/1 queue depends strongly on the form of the service time distribution, as we know from the Pollaczek-Khitchine formula. It is interesting to consider as well the M/D/1 queue. Here the influence of variations in service time are eliminated, and by [10, Theorem 8.6.2] we

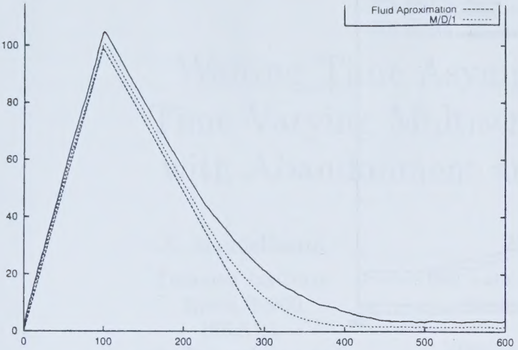


Figure 1: A single queue with a varying arrival rate

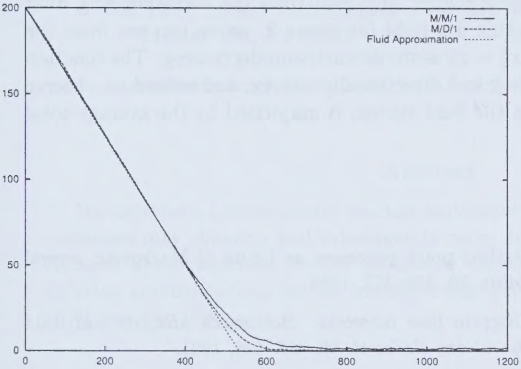


Figure 2: A single queue with a high initial workload

know that the workload in the M/D/1 queue is smaller than in the M/M/1 queue. (This result can also easily be established using the ideas of Section 4.) All three systems, the fluid approximation, the M/M/1 queue, and the M/D/1 queue, can be found in Figure 1.

In Figure 2 standard homogeneous M/M/1, M/D/1 queues and their fluid approximation are plotted, for arrival rate 0.5 and mean service time 0.8. The non-trivial factor is here the presence of a large initial workload. We see as in Figure 1 that the fluid model captures well the behavior of the queueing systems, as long as the stationary situation has not yet been reached.

Finally, in Figure 3, we consider a situation with two queues in tandem. It is a homogeneous system, where we assume a high initial load at the first queue, and a service rate at the second queue equal to 0.6, lower than the service rate at the first queue (0.8) (the inverse would give fluid limit 0 for the second queue all the time). In Figure 3 the workloads at queue 1 and 2 are plotted, both for the fluid model and the exponential model, as well as the sum of the two queues for each model. For reasons of clarity we did not add results for deterministic service times to the figure.

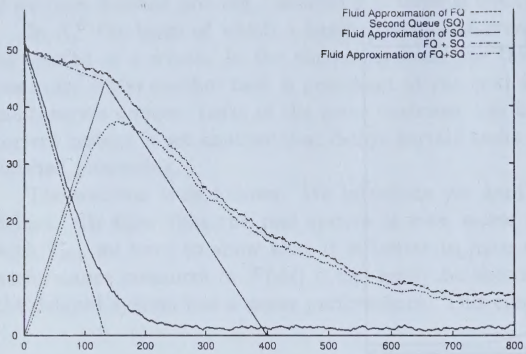


Figure 3: Tandem system with an initial workload and queue 1 faster than queue 2

Of course, queue 1 has on average a higher workload than the corresponding fluid queue. It is interesting to note that this not hold for queue 2, as we can see from the figure. This was to be expected, as $f(x) = x_2$ is not downstream decreasing. The function $f(x) = x_1 + x_2$ is downstream decreasing and directionally convex, and indeed we observe in Figure 3 that the total workload in the fluid system is majorized by the average total workload in the queueing system.

References

[1] S. Asmussen and G.M. Koole. Marked point processes as limits of Markovian arrival streams. *Journal of Applied Probability* **30**, 365–372, 1993.

[2] H. Chen and A. Mandelbaum. Discrete flow networks: Bottleneck analysis and fluid approximations. *Mathematics of Operations Research* **16**, 408–446, 1991.

[3] L. Kleinrock. *Queueing Systems, Volume II: Computer Applications.* Wiley, 1975.

[4] G.M. Koole. *Stochastic Scheduling and Dynamic Programming.* CWI, Amsterdam, 1995. CWI Tract 113.

[5] G.M. Koole. Structural results for the control of queueing systems using event-based dynamic programming. *Queueing Systems* **30**, 323–339, 1998.

[6] G.M. Koole. Convexity in tandem queues. Technical Report WS-516, Vrije Universiteit Amsterdam, 1999. (www.cs.vu.nl/~koole/papers/WS516.ps)

[7] G.M. Koole and Z. Liu. Stochastic bounds for queueing systems with multiple on-off sources. *Probability in the Engineering and Informational Sciences* **12**, 25–48, 1998.

[8] L.E. Meester and J.G. Shanthikumar. Regularity of stochastic processes. *Probability in the Engineering and Informational Sciences* **7**, 343–360, 1993.

[9] G.F. Newell. *Applications of Queueing Theory.* Chapman and Hall, 1971.

[10] S.M. Ross. *Stochastic Processes.* Wiley, 1983.

[11] R. Schassberger. *Warteschlangen.* Springer-Verlag, 1973.

Waiting Time Asymptotics for Time Varying Multiserver Queues with Abandonment and Retrials

A. Mandelbaum
Technion Institute
Haifa, 32000
ISRAEL
avim@tx.technion.ac.il

W. A. Massey
Bell Labs, Lucent Technologies
Murray Hill, NJ 07974
U.S.A.
will@research.bell-labs.com

M. I. Reiman
Bell Labs, Lucent Technologies
Murray Hill, NJ 07974
U.S.A.
marty@research.bell-labs.com

A. L. Stolyar
Bell Labs, Lucent Technologies
Murray Hill, NJ 07974
U.S.A.
stolyar@research.bell-labs.com

Abstract

We consider a nonstationary Markov multiserver queueing model where waiting customers may abandon and subsequently retry. In this paper we derive fluid and diffusion approximations for the associated *waiting time* process. The fluid and diffusion approximations for the corresponding *queue length* process were obtained in [4] (see also [5]).

1 Introduction

The model we consider in this paper is a multi-server queue with time-varying parameters, in which customers are impatient and hence abandon after (subjectively) excessive wait. Moreover, obtaining service is important enough for some customers that they return and seek service (retry) after a "time-out". Formally, our model is depicted in Figure 1: there is a single "service" node with n_t, $t \geq 0$, servers. New customers arrive to the service node following a Poisson process of rate λ_t. Customers arriving to find an idle server are taken into service that has rate μ_t^1. Customers that find all servers busy join a queue, from which they are served in a FCFS manner. Each customer waiting in the queue abandons at rate β_t. An abandoning customer leaves the system with probability ψ_t or joins a retrial pool with probability $1 - \psi_t$. Each customer in the retrial pool leaves to enter the service node at rate μ_t^2. Upon entry to the service node, these customers are treated the same as new customers. The behavior of the system is described by the two-dimensional, continuous time Markov chain $\mathbf{Q}(t) = \big(Q_1(t), Q_2(t)\big)$ where $Q_1(t)$ equals the number of customers residing in the service node (waiting or being served) and $Q_2(t)$ equals the number of customers in the retrial pool.

Our work is motivated by the need to develop analytical tools that support performance analysis of *large* telecommunication systems, such as telephone *call centers*, where

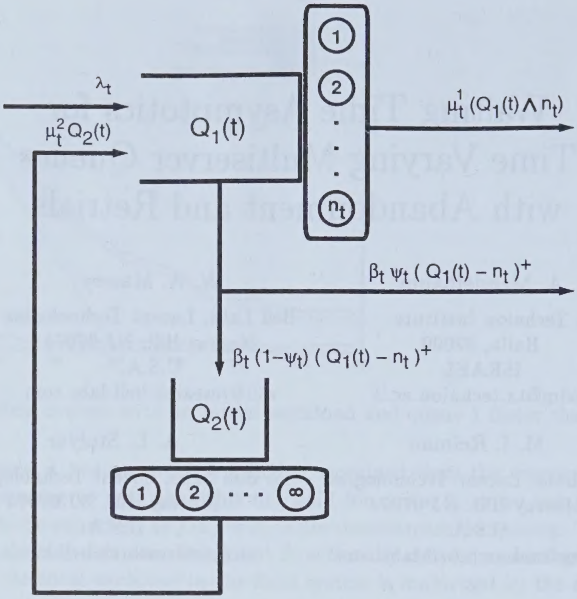

λ_t
$\mu_t^2 Q_2(t)$
$Q_1(t)$
1
2
n_t
$\mu_t^1 (Q_1(t) \wedge n_t)$
$\beta_t \psi_t (Q_1(t) - n_t)^+$
$\beta_t (1-\psi_t) (Q_1(t) - n_t)^+$
$Q_2(t)$
1
2
∞

Figure 1: The abandonment queue with retrials.

abandonments and retrials arise naturally. Call centers are constantly subject to time-varying conditions, and waiting customers in phone queues are unable to observe the state of the system. It follows that time-dependent modeling (as opposed to also state-dependent) is natural for call centers. Finally, we point out that the analysis of *waiting times* is (typically) analytically more challenging than that of the queue lengths, and in many applications (like call centers) is probably more important. For more discussion and related references on these issues, see [5].

The remainder of this paper is organized as follows. The asymptotic regime we consider is introduced in Section 2. In Section 3 we provide fluid and diffusion limits for the virtual waiting time at a fixed time τ. Process level fluid and diffusion limits for the virtual waiting time are presented in Section 4.

2 Asymptotic Regime

As mentioned above we are interested in the behavior of a system with large number of servers and large input rate. Thus, we consider the asymptotic regime where we scale up the number of servers in response to a similar scaling up of the arrival rate by customers.

More precisely, the asymptotic regime is as follows. Assume that $\lambda_t, \beta_t, \mu_t^1, \mu_t^2, \psi_t, n_t$ are fixed functions of time t. We consider a *sequence* of systems indexed by scaling parameter $\eta = \eta_1, \eta_2, \ldots, \eta_k \to \infty$ as $k \to \infty$. (To avoid cumbersome notation, in what follows, we index a system by η, and when we write $\eta \to \infty$, we mean that η goes to infinity by taking values from the sequence $\eta_1, \eta_2, \ldots.$) In a system with index η, the arrival rate (i.e., the intensity of the Poisson arrival process) is $\eta\lambda_t$ and the number of servers is ηn_t. (Actually, the latter should be, for example, the integer part of ηn_t, but again, to avoid trivial complications and simplify notation, we assume it's just ηn_t.) We also make the following additional

Assumption 2.1 The function n_t is *continuously differentiable* in $[0, \infty)$.

Sample paths of the family of queue length processes $\mathbf{Q}^\eta(t) = \left(Q_1^\eta(t), Q_2^\eta(t)\right)$, indexed by the scaling parameter η, are determined by the following equations:

$$\begin{aligned} Q_1^\eta(t) &= Q_1^\eta(0) + \Pi_{21}^c\left(\int_0^t Q_2^\eta(s)\mu_s^2 ds\right) - \Pi_{12}^b\left(\int_0^t \left(Q_1^\eta(s) - \eta n_s\right)^+ \beta_s(1-\psi_s)ds\right) \\ &\quad + \Pi^a\left(\int_0^t \eta\lambda_s ds\right) - \Pi^b\left(\int_0^t \left(Q_1^\eta(s) - \eta n_s\right)^+ \beta_s\psi_s ds\right) - \Pi^c\left(\int_0^t \left(Q_1^\eta(s) \wedge \eta n_s\right)\mu_s^1 ds\right) \end{aligned} \tag{2.1}$$

and

$$Q_2^\eta(t) = Q_2^\eta(0) + \Pi_{12}^b\left(\int_0^t \left(Q_1^\eta(s) - \eta n_s\right)^+ \beta_s(1-\psi_s)ds\right) - \Pi_{21}^c\left(\int_0^t \left(Q_2^\eta(s)\right)\mu_s^2 ds\right) , \tag{2.2}$$

where $\Pi^a, \Pi^b, \Pi^c, \Pi_{12}^b, \Pi_{21}^c$, are independent standard (rate 1) Poisson processes. In this paper we use the notation $x \wedge y = \min(x, y)$ and $x^+ = \max(x, 0)$ for all real x and y.

Throughout this paper we assume that the following *initial conditions* hold:

$$\frac{1}{\eta}\mathbf{Q}^\eta(0) \to \mathbf{Q}^{(0)}(0) , \tag{2.3}$$

$$\eta^{-1/2}[\mathbf{Q}^\eta(0) - \eta\mathbf{Q}^{(0)}(0)] \to \mathbf{Q}^{(1)}(0) , \tag{2.4}$$

where $\mathbf{Q}^{(0)}(0)$ and $\mathbf{Q}^{(1)}(0)$ are fixed vectors, and

$$Q_1^{(0)}(0) > 0 . \tag{2.5}$$

In the rest of the paper we also use the following notation. Let E be a complete separable metric space, and a be a real number. Then we denote by $\mathcal{D}(E, a)$ the Skorohod space of E-valued functions defined in the interval $[a, \infty)$ which are right continuous and have left limits. The space $\mathcal{D}(E, a)$ is endowed with Skorohod J_1-metric and the corresponding topology.

3 Waiting Time in Node 1: Marginal Distribution at a Given Time.

Suppose that we are interested in the waiting time of a "virtual" customer arriving at station 1 at a *fixed* time $\tau \geq 0$. Since we have a system with abandonment, a convenient way to approach this problem is to consider the system that is obtained from the original one by the following modification. *Suppose, that after time τ, there are no new exogenous arrivals into the system, and any customer departing any station i leaves the system.* In other words, starting time τ, each station i has no new arrivals, and it just serves the customers which were at the station at time τ. Theorem 5.1 in [4] still applies to the modified system; the only difference is that the terms in the equations, corresponding to the arrivals after time τ, should be "zeroed out". Namely, the following results follow directly from Theorem 5.1 (and its proof) in [4].

Denote the arrival and departure processes for station 1 by

$$A^\eta = \{\, A^\eta(t) \mid t \geq 0 \,\} \quad \text{and} \quad \Delta^\eta = \{\, \Delta^\eta(t) \mid t \geq 0 \,\}$$

respectively. Let, by convention, the arrival process include the customers in node 1 at time 0, so $A^\eta(0) = \mathbf{Q}_1^\eta(0)$, $\Delta^\eta(0) = 0$, and $A^\eta(t) - \Delta^\eta(t) = Q_1^\eta(t)$, $t \geq 0$.

Then we obtain the following *fluid* limit result.

Theorem 3.1 *With probability* 1, *the following convergence holds uniformly on compact sets (u.o.c.) of* t*:*

$$\frac{1}{\eta}(\mathbf{Q}^\eta, A^\eta, \Delta^\eta) \to (\mathbf{Q}^{(0)}, A^{(0)}, \Delta^{(0)}) \tag{3.1}$$

where $\mathbf{Q}^\eta = (Q_1^\eta, Q_2^\eta)$, $\mathbf{Q}^{(0)} = (Q_1^{(0)}, Q_2^{(0)})$, *the fluid limit* $Q^{(0)}$ *satisfies the following equations*

$$Q_1^{(0)}(t) = Q_1^{(0)}(0) + \int_0^t \left[\lambda_s + \mu_s^2 Q_2^{(0)}(s)\right] 1_{\{s \leq \tau\}} - \mu_s^1\big(Q_1^{(0)}(s) \wedge n_s\big) - \beta_s\big(Q_1^{(0)}(s) - n_s\big)^+ ds \tag{3.2}$$

and

$$Q_2^{(0)}(t) = Q_2^{(0)}(0) + \int_0^{t\wedge\tau} \beta_s(1-\psi_s)\big(Q_1^{(0)}(s) - n_s\big)^+ ds - \int_0^t \mu_s^2 Q_2^{(0)}(s) ds\,. \tag{3.3}$$

Moreover, $A^{(0)}$ *and* $\Delta^{(0)}$ *are equal to*

$$A^{(0)}(t) = Q_1^{(0)}(0) + \int_0^{t\wedge\tau} \left[\lambda_s + \mu_s^2 Q_2^{(0)}(s)\right] ds \tag{3.4}$$

and

$$\Delta^{(0)}(t) = \int_0^t \left[\mu_s^1\big(Q_1^{(0)}(s) \wedge n_s\big) + \beta_s\big(Q_1^{(0)}(s) - n_s\big)^+\right] ds\,, \tag{3.5}$$

where $\Delta^{(0)}$ *is a continuously differentiable non-decreasing function in* $[0, \infty)$.

We also obtain the following *diffusion* limit.

Theorem 3.2 *The following weak convergence holds (in the space being the direct product of corresponding Skorohod spaces* $\mathcal{D}(\mathbb{R}, 0)$*) :*

$$\sqrt{\eta}\Big(\frac{1}{\eta}\mathbf{Q}^\eta - \mathbf{Q}^{(0)}, \frac{1}{\eta}A^\eta - A^{(0)}, \frac{1}{\eta}\Delta^\eta - \Delta^{(0)}\Big) \xrightarrow{d} (\mathbf{Q}^{(1)}, A^{(1)}, \Delta^{(1)}), \tag{3.6}$$

where $\mathbf{Q}^{(1)} = (Q_1^{(1)}, Q_2^{(1)})$ *is the unique continuous solution to the stochastic differential equations*

$$\begin{aligned} Q_1^{(1)}(t) \;=\; & Q_1^{(1)}(0) + \int_0^t \left[\mu_s^1\big(Q_1^{(1)}(s) - Q_1^{(1)}(s)^*\big) + \beta_s Q_1^{(1)}(s)^*\right] ds \\ & + \int_0^{t\wedge\tau} \mu_s^2 Q_2^{(1)}(s) ds - B_{21}^c\Big(\int_0^{t\wedge\tau}\big(Q_2^{(0)}(s)\big)\mu_s^2 ds\Big) + B^a\Big(\int_0^{t\wedge\tau}\lambda_s ds\Big) \\ & - B_{12}^b\Big(\int_0^t \big(Q_1^{(0)}(s) - n_s\big)^+ \beta_s(1-\psi_s) ds\Big) - B^b\Big(\int_0^t \big(Q_1^{(0)}(s) - n_s\big)^+ \beta_s\psi_s ds\Big) \\ & - B^c\Big(\int_0^t \big(Q_1^{(0)}(s) \wedge n_s\big)\mu_s^1 ds\Big) \end{aligned} \tag{3.7}$$

and

$$\begin{aligned} Q_2^{(1)}(t) \;=\; & Q_2^{(1)}(0) + \int_0^{t\wedge\tau} Q_1^{(1)}(s)^* \beta_s(1-\psi_s) ds - \int_0^t \mu_s^2 Q_2^{(1)}(s) ds \\ & + B_{21}^c\Big(\int_0^{t\wedge\tau}\big(Q_2^{(0)}(s)\big)\mu_s^2 ds\Big) + B_{12}^b\Big(\int_0^{t\wedge\tau}\big(Q_1^{(0)}(s) - n_s\big)^+ \beta_s(1-\psi_s) ds\Big), \end{aligned} \tag{3.8}$$

with

$$Q_1^{(1)}(t)^* = Q_1^{(1)}(t)^+ 1_{\{Q_1^{(0)}(t)\geq n_t\}} - Q_1^{(1)}(t)^- 1_{\{Q_1^{(0)}(t)> n_t\}}, \tag{3.9}$$

where $A^{(1)}$ *and* $\Delta^{(1)}$ *are defined as*

$$A^{(1)}(t) = Q_1^{(1)}(0)+\int_0^{t\wedge\tau} \mu_s^2 Q_2^{(1)}(s)ds - B_{21}^c\left(\int_0^{t\wedge\tau}\left(Q_2^{(0)}(s)\right)\mu_s^2 ds\right)+B^a\left(\int_0^{t\wedge\tau}\lambda_s ds\right) \tag{3.10}$$

and

$$\begin{aligned}\Delta^{(1)}(t)=&\int_0^t \left[\mu_s^1\left(Q_1^{(1)}(s)-Q_1^{(1)}(s)^*\right)+\beta_s Q_1^{(1)}(s)^*\right]ds + B^c\left(\int_0^t\left(Q_1^{(0)}(s)\wedge n_s\right)\mu_s^1 ds\right) \\ &+B_{12}^b\left(\int_0^t\left(Q_1^{(0)}(s)-n_s\right)^+\beta_s(1-\psi_s)ds\right)+B^b\left(\int_0^t\left(Q_1^{(0)}(s)-n_s\right)^+\beta_s\psi_s ds\right).\end{aligned} \tag{3.11}$$

Clearly,

$$Q_1^{(1)}(t) = A^{(1)}(t) - \Delta^{(1)}(t)\ . \tag{3.12}$$

Now, let us define the "potential service initiation" process D^η for node 1 by

$$D^\eta(t) = \Delta^\eta(t) + \eta n_t,\ t\geq 0\ .$$

Note that if $Q_1^\eta(t) < \eta n_t$, then $A^\eta(t) < D^\eta(t)$; so the potential service can be "ahead" of arrivals.

Obviously, we have the (probability 1, u.o.c.) convergence:

$$\frac{1}{\eta}D^\eta(t)\to D^{(0)}(t),\ t\geq 0,$$

where $D^{(0)}(t) = \Delta^{(0)}(t) + n_t, t\geq 0$. Since n_t is continuously differentiable by assumption and we know that $\Delta^{(0)}(t)$ is continuously differentiable, $D^{(0)}(t)$ is also continuously differentiable and we denote its derivative by $d^{(0)}(t)$. Now we will make an important (but not very restrictive in majority of applications) additional assumption.

Assumption 3.1. The function $D^{(0)}$ (of t) is continuously differentiable with *strictly positive derivative*, and

$$\lim_{t\to\infty} D^{(0)}(t) > A^{(0)}(\tau)\ . \tag{3.13}$$

(Note, that according to our definitions, both $A^\eta(\cdot)$ and $A^{(0)}(\cdot)$ are constant in the interval $[\tau,\infty)$.)

Also, it will be convenient to adopt a convention that all the processes we consider are defined in the interval $[-T,\infty)$, with

$$T = n_0/d^{(0)}(0)\ .$$

We make this extension by assuming that nothing is happening in the interval $[-T,0)$ (no arrivals or departures) except the number of servers is increasing linearly from 0 to ηn_0 (for the unscaled process with index η).

We then can rewrite (3.1) and (3.6) as follows (with all the functions being now defined for $t\geq -T$):

$$\frac{1}{\eta}(\mathbf{Q}^\eta, A^\eta, D^\eta)\to(\mathbf{Q}^{(0)}, A^{(0)}, D^{(0)}) \tag{3.14}$$

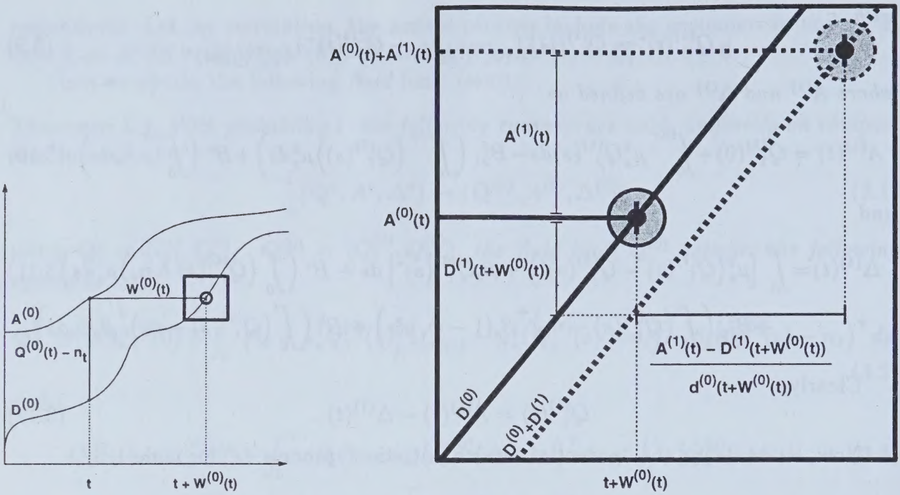


Figure 2: The diffusion term for the attainment waiting time

and

$$\sqrt{\eta}\left(\frac{1}{\eta}\mathbf{Q}^{\eta} - \mathbf{Q}^{(0)}, \frac{1}{\eta}A^{\eta} - A^{(0)}, \frac{1}{\eta}D^{\eta} - D^{(0)}\right) \xrightarrow{d} (\mathbf{Q}^{(1)}, A^{(1)}, D^{(1)}) \ , \tag{3.15}$$

where

$$D^{(1)} = \Delta^{(1)} . \tag{3.16}$$

Note that processes $A^{(0)}, D^{(0)}, A^{(1)}, D^{(1)}$ are continuous and $D^{(0)}(-T) = D^{(1)}(-T) = 0$.

Our conventions together with the Assumption 3.1 make the following processes well defined and finite with probability 1 for all sufficiently large η. Let us define, for all $t \geq -T$, the *first attainment* processes

$$S^{\eta}(t) = \inf\{s \geq -T : \ D^{\eta}(s) > A^{\eta}(t)\}$$

and

$$S^{(0)}(t) = \inf\{s \geq -T : \ D^{(0)}(s) > A^{(0)}(t)\}, \tag{3.17}$$

and the *attainment* waiting time processes

$$W^{\eta}(t) = S^{\eta}(t) - t$$

and

$$W^{(0)}(t) = S^{(0)}(t) - t \ . \tag{3.18}$$

Denote by $\hat{W}^{\eta}(\tau)$ the *virtual* waiting time at τ, i.e. the time a "test" customer (in the original non-modified system) arriving in node 1 at time τ would have to wait until its service starts, assuming this customer *does not abandon* while waiting. Then the relation between the virtual waiting time $\hat{W}^{\eta}(\tau)$ and the attainment waiting time $W^{\eta}(\tau)$ is simply

$$\hat{W}^{\eta}(\tau) = W^{\eta}(\tau)^{+} \ . \tag{3.19}$$

Indeed, note that $W^\eta(\tau)$ (and $W^{(0)}(\tau)$) may be negative. All this means is that $Q_1^\eta(\tau) < \eta n_\tau$, and therefore in this case $\hat{W}^\eta(\tau) = 0$. If $W^\eta(\tau)$ is non-negative, then its value is exactly equal to the virtual waiting time.

It follows directly from Theorem and Corollary in [7] that (3.14), (3.15), and Assumption 3.1, imply the following convergences.

With probability 1, u.o.c.,

$$(\frac{1}{\eta}\mathbf{Q}^\eta, \frac{1}{\eta}A^\eta, \frac{1}{\eta}D^\eta, W^\eta) \to (\mathbf{Q}^{(0)}, A^{(0)}, D^{(0)}, W^{(0)}) \ . \tag{3.20}$$

In distribution,

$$\sqrt{\eta}(\frac{1}{\eta}\mathbf{Q}^\eta - Q^{(0)}, \frac{1}{\eta}A^\eta - A^{(0)}, \frac{1}{\eta}D^\eta - D^{(0)}, W^\eta - W^{(0)}) \xrightarrow{d} (Q^{(1)}, A^{(1)}, D^{(1)}, W^{(1)}) \ , \tag{3.21}$$

where

$$W^{(1)}(t) = \frac{A^{(1)}(t) - D^{(1)}(S^{(0)}(t))}{d^{(0)}(S^{(0)}(t))} \ .$$

Since the processes $A^{(1)}, D^{(1)}, Q^{(1)}, W^{(1)}$ are continuous with probability 1, we automatically obtain the weak convergence of finite dimensional distributions.

In particular, consider the non-trivial case $S^{(0)}(\tau) \geq \tau$ (which is equivalent to $Q_1^{(0)}(\tau) \geq n_\tau$). We obtain

$$W^\eta(\tau) \to W^{(0)}(\tau)$$

and

$$\sqrt{\eta}(W^\eta(\tau) - W^{(0)}(\tau)) \xrightarrow{d} W^{(1)}(\tau) = \frac{Q_1^{(1)}(S^{(0)}(\tau))}{d^{(0)}(S^{(0)}(\tau))} \ .$$

Solving equation (3.2) for $Q_1^{(0)}(\cdot)$ in the interval $[\tau, \infty)$, we obtain

$$Q_1^{(0)}(t) = Q_1^{(0)}(\tau) \exp\left(-\int_\tau^t \beta_s ds\right) + \int_\tau^t \exp\left(-\int_s^t \beta_r dr\right)(\beta_s - \mu_s^1)n_s ds \ , \ t \geq \tau.$$

We can find $S^{(0)}(\tau)$ from

$$S^{(0)}(\tau) = \min\{t \geq \tau \mid Q_1^{(0)}(t) = n_t\} \ .$$

Solving a stochastic differential equation for $Q_1^{(1)}(\cdot)$ in the interval $[\tau, S^{(0)}(\tau)]$, we obtain (cf. [2]

$$Q_1^{(1)}(S^{(0)}(\tau)) \stackrel{d}{=} Q_1^{(1)}(\tau) \exp\left(-\int_\tau^{S^{(0)}(\tau)} \beta_s ds\right) + \int_\tau^{S^{(0)}(\tau)} \exp\left(-\int_s^t \beta_r dr\right) f_s dB(s - \tau) \ ,$$

where

$$f_t^2 = (Q_1^{(0)}(t) - n_t)\beta_t + n_t \mu_t^1 \ ,$$

and B is a standard Brownian motion process. In particular,

$$\mathsf{E}[Q_1^{(1)}(S^{(0)}(\tau))] = \mathsf{E}[Q_1^{(1)}(\tau)] \exp\left(-\int_\tau^{S^{(0)}(\tau)} \beta_s ds\right)$$

and

$$\mathsf{Var}[Q_1^{(1)}(S^{(0)}(\tau))] = \mathsf{Var}[Q_1^{(1)}(\tau)] \exp\left(-\int_\tau^{S^{(0)}(\tau)} 2\beta_r dr\right) + \int_\tau^{S^{(0)}(\tau)} \exp\left(-\int_s^{S^{(0)}(\tau)} 2\beta_r dr\right) f_s^2 ds \ .$$

Note that in case $Q_1^{(0)}(\tau) = n_\tau$, we obtain

$$S^{(0)}(\tau) = \tau,\ W^{(0)}(\tau) = 0,\ d^{(0)}(\tau) = \mu_\tau^1 n_\tau + n'_\tau,$$

and, therefore,

$$\sqrt{\eta} W^\eta(\tau) \xrightarrow{d} W^{(1)}(\tau) = \frac{Q_1^{(1)}(\tau)}{\mu_\tau^1 n_\tau + n'_\tau} .$$

Recalling (3.19), we obtain the following diffusion limit for the virtual waiting time in this case

$$\sqrt{\eta} \hat{W}^\eta(\tau) \xrightarrow{d} \frac{Q_1^{(1)}(\tau)^+}{\mu_\tau^1 n_\tau + n'_\tau}, \text{ if } Q_1^{(0)}(\tau) = n_\tau ,$$

which is what we intuitively expected.

We checked the accuracy of the fluid approximation for the virtual waiting time via simulation. The system we considered has all parameters constant except for λ_t. In particular we considered $n_t = 50$, $\mu_t^1 = 1$, $\mu_t^2 = 0.2$, $\beta_t = 0.25$, and $\psi_t = 0.5$, with $\lambda_t = 10 + 20t - t^2$, $0 \leq t \leq 20$. The results are shown in Figure 3. The graph on top compares the fluid and simulation results for the queue length, and the graph on the bottom compares the fluid and simulation results for the virtual waiting time. (The simulation results depicted are an average of 5000 independent replications. More details on the simulation method are contained in [5].)

4 Waiting Time in Node 1: A Process

In the previous section we derived fluid and diffusion approximations of the marginal distribution of the attainment waiting time, which uniquely determines those for the virtual waiting time, in node 1 *at a given time* $\tau \geq 0$. A natural conjecture is that one can obtain similar asymptotics for the attainment waiting time as a *random process* defined for $\tau \in [0, \infty)$. In this section we present results showing that the above conjecture is indeed true.

We need more definitions. First, in this section, unless otherwise explicitly stated, we view all the processes as random processes of two time variables, $t \in [-T, \infty)$ and $\tau \in [0, \infty)$. (In the previous section τ was a fixed parameter.) More precisely, we view them as random elements $X = ((X(t,\tau),\ t \in [-T, \infty)),\ \tau \geq 0)$ (X can be Q_i^η or A^η or $Q_i^{(j)}$, etc.) taking values in the space $\mathcal{D}(\mathcal{D}(\mathbb{R}, -T), 0)$.

Note that for each fixed τ all processes of interest are well defined in the previous section, and the convergences (3.20) and (3.21) do hold for any fixed τ.

Assumption 4.1 Assumption 3.1 holds for any $\tau \geq 0$.

A generalization of the argument used in the proofs in [4] (roughly, making all estimates in the convergence proofs "uniform on τ"), and a generalization of the results in [7], lead to the following results which are extensions of (3.20) and (3.21). The details are contained in [6]. First, we state our functional strong law of large numbers result.

Theorem 4.1 *With probability* 1, *uniformly on compact sets of* (t, τ),

$$(\frac{1}{\eta}\mathbf{Q}^\eta, \frac{1}{\eta}A^\eta, \frac{1}{\eta}D^\eta, S^\eta, W^\eta) \to (\mathbf{Q}^{(0)}, A^{(0)}, D^{(0)}, S^{(0)}, W^{(0)}) , \tag{4.1}$$

where all functions $\mathbf{Q}^{(0)}$, $A^{(0)}$, $D^{(0)}$, $S^{(0)}$, $W^{(0)}$, *are continuous jointly on* τ *and* t, *and for each fixed* τ *they (as functions of* t*) satisfy the ODE (3.2), (3.3), and equations (3.4), (3.5), (3.17), (3.18). Moreover,* $d^{(0)}(t,\tau) \equiv (\partial/\partial t) D^{(0)}(t,\tau)$ *is strictly positive.*

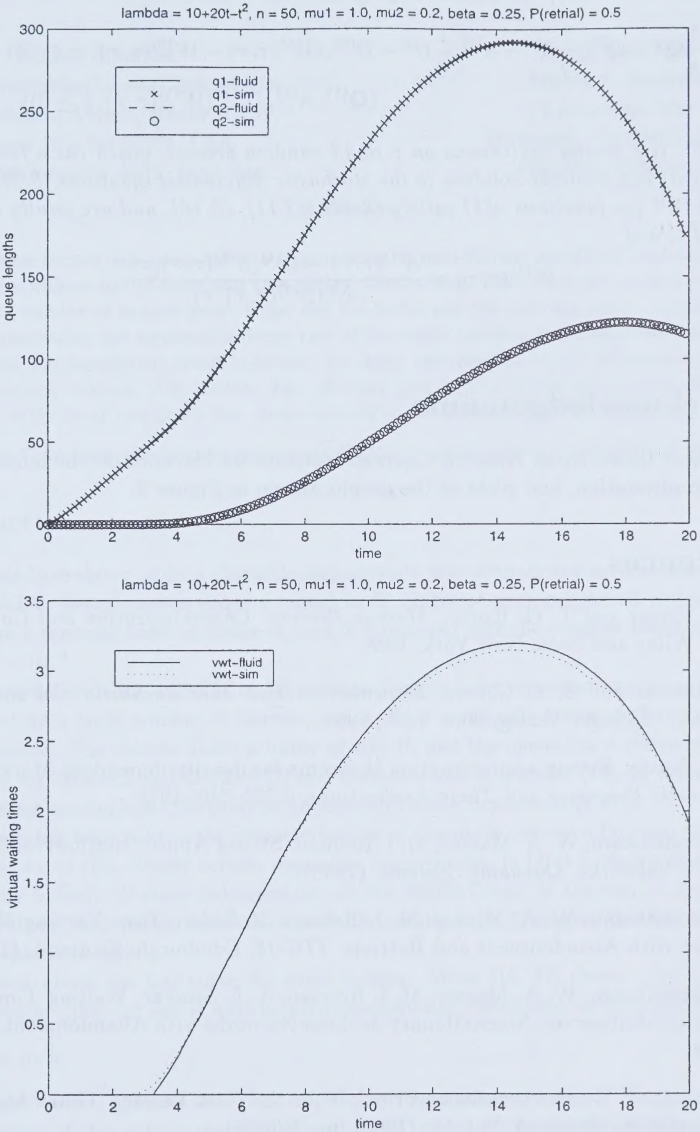


Figure 3: The fluid approximations for the queue lengths and virtual waiting time

Now we state our functional central limit theorem.

Theorem 4.2 *The following weak convergence holds:*

$$\sqrt{\eta}\left(\frac{1}{\eta}\mathbf{Q}^{\eta} - \mathbf{Q}^{(0)}, \frac{1}{\eta}A^{\eta} - A^{(0)}, \frac{1}{\eta}D^{\eta} - D^{(0)}, (W^{\eta}(\tau,\tau) - W^{(0)}(\tau,\tau),\ \tau \geq 0)\right) \xrightarrow{d} \left(\mathbf{Q}^{(1)}, A^{(1)}, D^{(1)}, (W^{(1)}(\tau,\tau), \tau \geq 0)\right), \tag{4.2}$$

where $\mathbf{Q}^{(1)}$ *is a jointly continuous on* τ *and* t *random process, which (as a function of* t*, with* τ *fixed) is the unique solution to the stochastic differential equations (3.7) and (3.8);* $A^{(1)}$ *and* $D^{(1)}$ *(as functions of* t*) satisfy (3.10), (3.11), (3.16), and are jointly continuous on* τ *and* t*; and*

$$W^{(1)}(\tau,\tau) = \frac{A^{(1)}(\tau,\tau) - D^{(1)}(S^{(0)}(\tau,\tau),\tau)}{d^{(0)}(S^{(0)}(\tau,\tau),\tau)}$$

is continuous on τ*.*

5 Acknowledgements

The authors thank Brian Rider of Courant Institute for his work on the numerical simulation, computation, and plots of the graphs shown in Figure 3.

References

[1] S. N. Ethier and T. G. Kurtz. *Markov Process: Characterization and Convergence.* John Wiley and Sons, New York, 1986.

[2] I. Karatzas and S. E. Shreve. *Brownian Motion and Stochastic Calculus (Second Edition).* Springer-Verlag, New York, 1991.

[3] T. G. Kurtz. Strong approximation theorems for density dependent Markov chains. *Stochastic Processes and Their Applications*, 6:223–240, 1978.

[4] A. Mandelbaum, W. A. Massey, M. I. Reiman. Strong Approximations for Markovian Service Networks. *Queueing Systems*, (1998).

[5] A. Mandelbaum, W. A. Massey, M. I. Reiman, B. Rider. Time Varying Multiserver Queues with Abandonment and Retrials. *ITC-16*, Edinburgh, Scotland, (1999).

[6] A. Mandelbaum, W. A. Massey, M. I. Reiman, A. L. Stolyar. Waiting Time Asymptotics for Multiserver, Nonstationary Jackson Networks with Abandonment. *In preparation.*

[7] A. Puhalskii. On the Invariance Principle for the First Passage Time. *Mathematics of Operatioins Research*, Vol. 19, (1994), pp. 946-954.

Large deviations for small buffers: an insensitivity result

Michel Mandjes[1]
Bell Laboratories/Lucent Technologies
600 Mountain Avenue, Room 2C361
Murray Hill, NJ 07974, USA.
`michel@research.bell-labs.com`

Jeong Han Kim
Microsoft Research
1 Microsoft Way
Redmond, WA 98052, USA.
`jehkim@microsoft.com`

Abstract

This article focuses on a queue fed by a large number of 'semi-Markov modulated fluid sources', e.g., on/off sources with on and off-times that have general distributions. The asymptotic regime is considered in which the number of sources grows large, and the buffer and link rate are scaled accordingly. We aim at characterizing the exponential decay rate of the buffer overflow probability for the regime of small buffers. An insensitivity result is proven: the decay rate depends on the distributions of the on and off-times only through their means. The efficiency gain to be achieved by using small buffers is significant, as the decay rate grows fast: proportionally to the square root of the buffer size.

Keywords: Fluid sources — large deviations asymptotics — small buffers — insensitivity — ATM/IP.

Introduction

tochastic models have shown to be a tractable and accurate way of analyzing modern telecommunication etworks, e.g. ATM (asynchronous transfer mode) or IP (Internet protocol) based systems. In practice ıany flows share a common buffered resource, and it is required that the overflow probability is typically mall, like $10^{-6} - 10^{-8}$.

A common way of modeling an ATM or IP network element is by representing the system as a queue. he queue is fed by a large number of sources, say n. It is commonly assumed that these sources are atistically identical. The sources share a buffer of size B, and the queue has a constant depletion rate . As shown by e.g. Botvich and Duffield [2] and Likhanov and Mazumdar [12], by scaling $B \equiv nb$ and $= nc$, under weak assumptions the overflow probability decays exponentially in n.

Much attention has been paid to the characterization of this decay rate as a function of the buffer size, ıe so-called loss curve $I(b)$. Under certain regularity requirements, in [2] it is shown that $I(b)$ grows for ırge b essentially linearly. If these requirements are not fulfilled (e.g., in the case of on/off traffic with eavy-tailed on-times) only partial results are available; results in [6, 7] suggest that for regularly varying n-times $I(b)$ behaves as $\log b$.

Little is known about the loss curve for small buffers. Weiss [15, 17] shows – by a pathwise large eviations approach – that for on/off sources with exponentially distributed on and off-times

$$I(b) \approx \alpha + \beta\sqrt{b}; \tag{1}$$

ourcoubetis and Weber [4] report that they empirically found this property.

The contribution of this note is the extension of Weiss' result (1) to a much broader class of sources: -called *semi-Markov modulated fluid sources.* The traffic rate generated by a single source alternates between a (finite) number of constant traffic rates; the sojourn time at any traffic rate has a general distribution and the transitions between the traffic rates are Markovian. The establishment of (1) is done by exploiting the characterization of the decay rate given by Likhanov and Mazumdar [12].

---

[1] Corresponding author.

Strikingly, for small b loss curve $I(b)$ only depends on the distribution of the sojourn times through their means. This *insensitivity* result has useful practical consequences. Loosely speaking it implies that for traffic management purposes only the means should be taken into account, and knowledge of the entire distributions is not required. Also, due to the square root the loss curve is steep for small b; this means that adding a small buffer causes a sharp reduction of the overflow probability.

The organization of this article is as follows. Section 2 gives the model. In Section 3 we present our main theorem. We also comment on the small buffer asymptotics for an open model; here the number of sources is not constantly n, but rather has an M/G/∞ type of distribution. Section 4 comments on the practical implications of the asymptotic relation (1).

2 Model

In this note we will concentrate on a type of sources that we call *semi-Markov modulated fluid sources.* Such a source can be described by (i) a stochastic process $X(t)$ (where $t \geq 0$) attaining values in $\{1, \ldots, d\}$, the so-called *modulating process*, and (ii) a vector of positive and finite *traffic rates* $r_1, \ldots, r_d$.

The modulating process stays in state $i \in \{1, \ldots, d\}$ during a sojourn time with distribution S_i. Denoting by $S_{i,j}$ the duration of the jth visit to state i, $(S_{i,j})_{j=1}^{\infty}$ is a sequence of i.i.d. random variables. Also $(S_{i_1,j})_{j=1}^{\infty}$ and $(S_{i_2,j})_{j=1}^{\infty}$ are independent, $i_1 \neq i_2$.

When leaving state i the probability of going to j is p_{ij}, independent of the sojourn times. It is assumed that the (discrete-time) Markov chain with transition matrix $(p_{ij})_{i,j=1}^{d}$ is irreducible, and consequently it has a unique invariant distribution, say ρ. When the modulating process is in state i, the source generates traffic at a constant rate r_i. We define $A(t)$ as the (steady-state) amount of traffic generated by a single source in $[0, t]$. We denote the source's peak rate $\max_i r_i$ by r.

Later in our analysis we need the following assumptions.

Assumption 2.1 *For all* $i \in \{1, \ldots, d\}$ *the mean sojourn time* $\mathbb{E}S_i = \mu_i$ *is finite. The distribution of* $\sum_{i=1}^{d} S_i$ *is non-lattice.*

Assumption 2.2 *The probability that the modulating process makes two or more transitions in an interval of length* t *is* $O(t^2)$.

The first assumption has two major implications – for details we refer to Section 2.1 of [7]. In the first place, it assures that the long-run fraction of time the source spends in state i is

$$\pi_i := \frac{\rho_i \mu_i}{\sum_j \rho_j \mu_j}.$$

Also the residual sojourn time is well defined: observing that a modulating process is in state i, the residual sojourn time S_i^* has distribution

$$\mathbb{P}(S_i^* > x) = \mu_i^{-1} \int_x^{\infty} \mathbb{P}(S_i > y) \mathrm{d}y.$$

The second assumption enables us to generate a linear approximation of the moment generating function of $A(t)$, see the proof of Theorem 3.3. The assumption is satisfied in nearly all practical cases; a sufficient condition is that the corresponding density is bounded.

We assume that there are n sources $(X(\cdot), r)$, that feed into a queue with infinite buffer and constant k rate C. We focus on the probability that the buffer level exceeds a (low) level B. Then the resources rescaled: $C \equiv nc$ and $B \equiv nb$. We assume that the system is stable: $\sum_j \pi_j r_j < c$.

The performance measure we are interested in is the *overflow probability*, or more precisely stated:

$p_n(b,c)$ = steady-state probability that the buffer content exceeds level nb.

in particularly focus on its exponential *decay rate* (as a function of b, for fixed c):

$$I(b) := -\lim_{n\to\infty} \frac{1}{n} \log p_n(b,c).$$

r main theorem describes this decay rate for small b.

Analysis

this section we state and prove our main result: the shape of the loss curve $I(b)$ for small values of b. is is done in Theorem 3.3. Its proof is based on a variant of the key theorem of Likhanov and Mazumdar], that we establish in Theorem 3.2. It should be mentioned that results related to Theorem 3.2 were ven by Botvich and Duffield [2], Courcoubetis and Weber [4], and Simonian and Guibert [16]. Theorem requires the following additional assumption.

sumption 3.1 *Define*

$$J_t(b) := \sup_{\theta} \left(\theta(b+ct) - \log \mathbb{E} e^{\theta A(t)} \right).$$

i) Assume that for any $b \geq 0$, $\liminf_{t\to\infty} J_t(b)(\log t)^{-1} > 0$;

i) Assume that $J(b) := \inf_{t>0} J_t(b)$ *is a continuous function of* b.

eorem 3.2 [Loss curve for general b**]** *Under Assumptions 2.1 and 3.1,*

$$I(b) := -\lim_{n\to\infty} \frac{1}{n} \log p_n(b,c) = J(b). \tag{2}$$

oof. The proof consists of a lower and upper bound. The lower bound mimics the line of reasoning of tvich and Duffield [2]. The upper bound relies on the result of Likhanov and Mazumdar [12], who prove imilar property in a discrete-time setting.

wer bound. First recall the standard result that the stationary workload has the same distribution as $_{t>0} \sum_{i=1}^{n} A_i(t) - nct$, where $A_i(t)$ is the (steady-state) amount of traffic generated by source i in an erval of length t. Then

$$\begin{aligned}
\liminf_{n\to\infty} \frac{1}{n} \log p_n(b,c) &\overset{\text{(i)}}{=} \liminf_{n\to\infty} \frac{1}{n} \log \mathbb{P}\left(\sup_{t>0} \sum_{i=1}^{n} A_i(t) - nct > nb \right) \\
&\overset{\text{(ii)}}{\geq} \liminf_{n\to\infty} \frac{1}{n} \sup_{t>0} \log \mathbb{P}\left(\sum_{i=1}^{n} A_i(t) - nct > nb \right) \\
&\overset{\text{(iii)}}{\geq} \sup_{t>0} \liminf_{n\to\infty} \frac{1}{n} \log \mathbb{P}\left(\sum_{i=1}^{n} A_i(t) - nct > nb \right) \\
&\overset{\text{(iv)}}{\geq} \sup_{t>0} \left(-J_t(b^+) \right) \overset{\text{(v)}}{=} -J(b).
\end{aligned}$$

Here Equation (i) is by definition. Inequality (ii) uses that the probability of the union of events is larger than the probability of any of these events. Inequality (iii) is standard. Then we apply the lower bound of the Gärtner-Ellis theorem in Inequality (iv); see p. 45 of Dembo and Zeitouni [5]. For the Gärtner-Ellis theorem to hold, it is noted that for any t it holds that $\log \mathbb{E}\exp(\theta A(t)) < \infty$, Assumption 2.3.2 of [5]. Equality (v) is due to the continuity of $J(b)$, i.e., Assumption 3.1.(ii).

Upper bound. First we derive the desired property in discrete time; with Assumption 3.1.(ii) we translate it into continuous time.

First define the overflow probability in an 'ϵ-grid':

$$p_n^{\epsilon}(b,c) := \mathbb{P}\left(\sup_{t\in\epsilon\mathbb{N}} \sum_{i=1}^{n} A_i(t) - nct > nb\right),$$

where $\epsilon\mathbb{N} := \{\epsilon, 2\epsilon, \ldots\}$. Under Assumption 2.1 (implying that the sources are stationary) and Assumption 3.1.(i), Likhanov and Mazumdar [12] have shown

$$\lim_{n\to\infty} \frac{1}{n} \log p_n^{\epsilon}(b,c) = -\inf_{t\in\epsilon\mathbb{N}} J_t(b). \tag{3}$$

In fact they even find a subexponential function $K(n)$ such that for $n\to\infty$,

$$p_n^{\epsilon}(b,c)K(n)\exp\left(n \inf_{t\in\epsilon\mathbb{N}} J_t(b)\right) \to 1.$$

To prove this they need the additional assumption that $\inf_{t\in\epsilon\mathbb{N}} J_t(b)$ has a unique optimizing argument t_0. However, from their proof it is easy to see that the weaker statement (3) does not require this uniqueness.

Notice that the fact that the peak rate r is bounded implies that, for all positive ϵ, we have $p_n(b,c) \le p_n^{\epsilon}(b-r\epsilon, c)$. Consequently, for all $\epsilon > 0$,

$$\limsup_{n\to\infty} \frac{1}{n} \log p_n(b,c) \le -\inf_{t\in\epsilon\mathbb{N}} J_t(b-r\epsilon) \le -\inf_{t>0} J_t(b-r\epsilon) = J(b-r\epsilon). \tag{4}$$

Due to the continuity of $J(\cdot)$ (Assumption 3.1(ii)) the last expression converges to $J(b)$ when $\epsilon \downarrow 0$. □

Before stating our main theorem, we first introduce some notation. For zero buffers the decay rate $I(0)$ can be found by applying the classical result of Cramér [15]. It is given by

$$\alpha := \sup_{\theta}\left(\theta c - \log \mathbb{E}e^{\theta r_{X(0)}}\right), \tag{5}$$

cf. Hui [10] and Theorem 2 of Botvich and Duffield [2].

Let θ_0 be the optimizing argument in the right hand side of (5). Differentiating $\theta c - \log \mathbb{E}e^{\theta r_{X(0)}}$ to θ, and equating the derivative to zero, we see that θ_0 solves

$$\frac{\mathbb{E}r_{X(0)}e^{\theta_0 r_{X(0)}}}{\mathbb{E}e^{\theta_0 r_{X(0)}}} = \frac{\sum_j \pi_j r_j e^{\theta_0 r_j}}{\sum_j \pi_j e^{\theta_0 r_j}} = c; \tag{6}$$

it is well known that (6) has a unique solution, due to the convexity of $\log \mathbb{E}e^{\theta r_{X(0)}}$. We define β as follows:

$$a_j := -\frac{e^{\theta_0 r_j}}{\mu_j} + \sum_{k=1}^{d} \frac{p_{jk}}{\mu_j} \cdot \frac{e^{\theta_0 r_j} - e^{\theta_0 r_k}}{\theta_0(r_j - r_k)}, \quad \beta := 2\left(\frac{\sum_j \pi_j \theta_0 a_j}{\sum_j \pi_j e^{\theta_0 r_j}}\right)^{\frac{1}{2}}. \tag{7}$$

heorem 3.3 [loss curve for small b] *With α and β defined by (5) and (7), for small b,*

$$\lim_{n\to\infty} \frac{1}{n} \log p_n(b,c) = -\alpha - \beta\sqrt{b} + O(b), \tag{8}$$

nder Assumptions 2.1, 2.2, and 3.1.

roof. The proof is based on Theorem 3.2, and consists of an upper and a lower bound.

pper bound. From (2)

$$\limsup_{n\to\infty} \frac{1}{n} \log p_n(b,c) \leq -\inf_{t>0} \left(\theta(t)(b+ct) - \log \mathbb{E} e^{\theta(t)A(t)} \right), \tag{9}$$

or any function $\theta(\cdot)$. We here choose $\theta(t) = \theta_0/t$. First we evaluate the first two terms of the Taylor series f $\mathbb{E}\exp(\theta_0 A(t)/t)$, $t \downarrow 0$. Invoking Assumption 2.1,

$$\mathbb{E}\left(e^{\theta_0 A(t)/t}\right) = \sum_{j=1}^{d} \pi_j \mathbb{E}_j e^{\theta_0 A(t)/t}, \text{ where } \mathbb{E}_j e^{\theta_0 A(t)/t} := \mathbb{E}\left(e^{\theta_0 A(t)/t} \mid X(0) = j\right).$$

ased on the fact that $A(t) \leq rt$, and taking into account only constant and linear (in t) terms,

$$\begin{aligned} \mathbb{E}_j e^{\theta_0 A(t)/t} &= e^{\theta_0 r_j} \mathbb{P}(S_j^* > t) + \sum_{k=1}^{d} \int_0^t \mathbb{P}(S_j^* = x) p_{jk} \exp\left(\theta_0 \left(r_j \frac{x}{t} + r_k \frac{t-x}{t}\right)\right) \mathrm{d}x \\ &= e^{\theta_0 r_j}\left(1 - \frac{t}{\mu_j}\right) + \sum_{k=1}^{d} \frac{p_{jk} t}{\mu_j} \cdot \frac{e^{\theta_0 r_j} - e^{\theta_0 r_k}}{\theta_0 (r_j - r_k)} = e^{\theta_0 r_j} + a_j t. \end{aligned} \tag{10}$$

ue to Assumption 2.2 – the probability of two or more transitions in an interval of length t is $O(t^2)$ – the ght hand side of (9) reads

$$-c\theta_0 + \log \sum_{j=1}^{d} \pi_j e^{\theta_0 r_j} - \inf_{t>0} \left(\frac{b\theta_0}{t} + t \frac{\sum_j \pi_j a_j}{\sum_j \pi_j e^{\theta_0 r_j}} + O(t^2) \right). \tag{11}$$

eglecting the $O(t^2)$ term, the unique infimum is assumed for

$$t = K\sqrt{b}, \text{ with } K := \left(\frac{\sum_j \pi_j \theta_0 e^{\theta_0 r_j}}{\sum_j \pi_j a_j} \right)^{\frac{1}{2}}; \tag{12}$$

he possible error in (11) is $O((K\sqrt{b})^2) = O(b)$. Inserting $t = K\sqrt{b}$ into (11) we get the upper bound:

$$\limsup_{n\to\infty} \frac{1}{n} \log p_n(b,c) \leq -\alpha - \beta\sqrt{b} + O(b).$$

ower bound. Again from (2), for any t^*,

$$\liminf_{n\to\infty} \frac{1}{n} \log p_n(b,c) \geq -\sup_{\theta} \left(\theta(b+ct^*) - \log \mathbb{E} e^{\theta A(t^*)} \right) = -\sup_{\theta} \left(\frac{\theta b}{t^*} + c\theta - \log \mathbb{E} e^{\theta A(t^*)/t^*} \right). \tag{13}$$

he right hand side of (13) has a unique optimizer, as follows from the convexity of the log moment enerating function. As suggested by Equation (12) we choose $t^* := K\sqrt{b}$ in (13). To find the supremum e differentiate the supremand of (13) and equate it to zero. This gives

$$\frac{\sqrt{b}}{K} + c = \frac{\mathbb{E}(K\sqrt{b})^{-1} A(K\sqrt{b}) e^{\theta(K\sqrt{b})^{-1} A(K\sqrt{b})}}{\mathbb{E} e^{\theta(K\sqrt{b})^{-1} A(K\sqrt{b})}}. \tag{14}$$

The numerator and denominator of the right hand side of (14) can be written as (η being a fixed number)

$$\mathbb{E}r_{X(0)}e^{\theta r_{X(0)}} + \eta\sqrt{b} + O(b), \text{ and } \mathbb{E}e^{\theta r_{X(0)}} + O(\sqrt{b}),$$

respectively. This is done similarly to the derivation of (10); the $O(b)$ and $O(\sqrt{b})$ terms are due to Assumption 2.2. Hence

$$c = \frac{\mathbb{E}r_{X(0)}e^{\theta r_{X(0)}}}{\mathbb{E}e^{\theta r_{X(0)}}} + \frac{\eta}{\mathbb{E}e^{\theta r_{X(0)}}}\sqrt{b} + O(b), \tag{15}$$

Calling the θ that solves Equation (14) $\theta_0(b)$, we know from (15) that $\theta_0(b) = \theta_0 + O(\sqrt{b})$, cf. Eq. (6).

Because functions are quadratic around their maximum, the right hand side of (13) reads

$$-c\theta_0 - \frac{\theta_0 b}{K\sqrt{b}} - \log \mathbb{E}\exp\left(\theta_0 (K\sqrt{b})^{-1} A(K\sqrt{b})\right) + O(b).$$

Applying the linear expansion of $\mathbb{E}\exp(\theta_0 A(t)/t)$ around zero, as was derived in the upper bound – see (10) – we get

$$-c\theta_0 + \log \sum_{j=1}^{d} \pi_j e^{\theta_0 r_j} - \frac{\theta_0\sqrt{b}}{K} - K\sqrt{b}\frac{\sum_j \pi_j a_j}{\sum_j \pi_j e^{\theta_0 r_j}} + O(b) \tag{16}$$

Inserting the specific form of K – see (12) – into expression (16) gives lower bound

$$\liminf_{n\to\infty} \frac{1}{n} \log p_n(b,c) \leq -\alpha - \beta\sqrt{b} + O(b).$$

□

Remark: discussion on Assumption 3.1. As Assumption 3.1.(ii) is merely technical, we will pay some more attention to (i). As proven in Proposition 3.1 of Likhanov and Mazumdar [12], a sufficient condition for Assumption 3.1.(ii) to hold is that for t large enough $\mathbb{P}(A(t) > (\mu+\epsilon)t) \leq Kt^{-\alpha}$, for positive constants ϵ, K, α, and μ defined as the source's mean rate. Likhanov and Mazumdar argue that this condition is fulfilled for a broad class of on-off sources. In particular, it holds for sources with a *heavy-tailed* on-time, i.e., regularly varying with finite mean and infinite variance.

This explains why we did not use the results of Botvich and Duffield [2], but rather our continuous-time version of Likhanov and Mazumdar [12]. Botvich and Duffield [2] prove relation (2) under different assumptions. In particular, they require that there is a θ such that for t large enough $\log \mathbb{E}e^{\theta A(t)} < c\theta t$. It is not hard to show that this requirement is not fulfilled for on-off sources with heavy-tailed on-time. This is done as follows.

Denote by π_{on} the steady-state on-probability; let r be the traffic rate of the on-state (larger than c). Assume that the on-time has a complementary distribution function that behaves like $t^{-\epsilon}$, for $\epsilon > 0$. Then, for $\theta > 0$ and t large:

$$\mathbb{E}e^{\theta A(t)} \geq \pi_{\text{on}} t^{-\epsilon} e^{\theta r t} \geq e^{\theta c t};$$

here we focused on the probability that the source is on at time 0 and stays on during $[0,t]$. For ϵ between 1 and 2 we get finite mean and infinite variance, i.e., the heavy-tailed regime.

Remark: on-off sources. For on-off sources α and β can be calculated explicitly. Let μ_{on} denote the mean on-time and μ_{off} the mean off-time; r is the traffic rate while the source is on. Then the solution of (6) is

$$\theta_0 = \frac{1}{r}\log\left(\frac{c}{r-c}\cdot\frac{\mu_{\text{off}}}{\mu_{\text{on}}}\right).$$

This leads to

$$\alpha = \frac{c}{r}\log\left(\frac{c}{r}\cdot\frac{\mu_{\text{on}}+\mu_{\text{off}}}{\mu_{\text{on}}}\right)+\left(1-\frac{c}{r}\right)\log\left(\left(1-\frac{c}{r}\right)\cdot\frac{\mu_{\text{on}}+\mu_{\text{off}}}{\mu_{\text{off}}}\right),\ \text{and}$$

$$\beta = 2\sqrt{\left(\frac{c}{\mu_{\text{on}}}+\frac{r-c}{\mu_{\text{off}}}\right)\log\left(\frac{c}{r-c}\cdot\frac{\mu_{\text{off}}}{\mu_{\text{on}}}\right)-2\left(\frac{c}{\mu_{\text{on}}}-\frac{r-c}{\mu_{\text{off}}}\right)}.$$

As argued in the previous remark, these formulae – found by Weiss [17] for exponential on-off traffic – hold for a broad class of on-off sources.

4 Practical implications

Insensitivity. The exponential decay rate depends on the distribution of the sojourn times S_i, $i = \{1,\dots,d\}$, only through their means $(\mu_i)_{i=1}^d$. This has two important practical implications:
(1) Some recent literature has advocated the use of traffic models that exhibit *long-range dependence.* This is motivated by studies on network traffic in e.g. Ethernet Local Area Networks (see Leland *et al.* [11]) and Wide Area Networks (see Paxson and Floyd [13]).

One way of modeling long-range dependence is by taking on/off sources with heavy-tailed on-times, e.g., regularly varying tails with finite mean and infinite variance, see Boxma and Dumas [3]. It was shown that in the regime with *large* buffers these sources cause overflow behavior that is essentially different from the situation with exponentially (or phase-type) distributed on and off-times. Roughly, in the former case the overflow probability decays polynomially [7], in the latter case exponentially [1].

Theorem 3.3 states that for *small* buffers and a large number of sources there is not such a fundamental difference. For instance for on/off sources, only the *mean* on and off-times are relevant; it is not important whether there are heavy tails or exponential tails. In other words: the small buffer results found by Weiss [17] for exponential on/off sources apply for general on/off sources (with the same mean on and off-times).

It should be mentioned that the same properties are still valid when considering the loss probability in a finite-buffer system (rather than the buffer overflow probability in an infinite-buffer system), cf. [4, 12]. Then the insensitivity result can be considered as complementary to the studies of Ryu and Elwalid [14] and Heyman and Lakshman [9]. There it is argued that – even in the presence of long-range dependent traffic – for realistic values of the maximum delay (and corresponding buffer sizes), a Markovian (short-range dependent) modeling accurately predicts the loss probability.
(2) Theorem 3.3 enables approximation of loss probabilities in a queue (of small size) fed by *worst-case traffic*, i.e., traffic with deterministic on and off-times.

In the literature significant attention has been paid to this type of 'adversarial traffic'. In ATM and IP networks traffic is usually characterized by a three-parameter traffic descriptor. The worst case traffic that fits within this descriptor (in the sense that it maximizes the overflow probability) is on-off traffic with fixed on and off times. Notice that if n of these sources are multiplexed, the only random effect is their phase: the start of the on-time is uniformly distributed on the period of the source.

Analytically, this model is notoriously hard to analyze. A practically useful conservative solution was given by [8]. Theorem 3.3 gives an accurate approximation for the case of many sources and small buffers.

Loss curve. For small buffers the loss curve $I(b)$ increases like $\sqrt{b}$. In other words: for small buffers the overflow probability decreases very fast, so there is a large 'marginal benifit' of an additional unit of buffer space. It is emphasized that this marginal benifit may be smaller for large b. If the on-times have an exponentially bounded distribution $I(b)$ is essentially linear; the results of Duffield [6] and Dumas and Simonian [7] suggest that for on-times with a regularly varying distribution the loss curve even looks like $\log b$ for large b.

It was already widely recognized that small buffers were useful to absorb traffic fluctuations at the packet level. The queue is then used to cope with the asynchronous arrival of information packets. However, the shape of the loss curve for small b states that even if traffic is modeled as fluid (and it consequently ignores the packet level), it is worthwhile to have a small buffer.

References

[1] D. Anick, D. Mitra, and M. Sondhi. Stochastic theory of a data-handling system with multiple sources. *The Bell System Technical Journal*, 61: 1871 – 1894, 1982.

[2] D. Botvich and N. Duffield. Large deviations, the shape of the loss curve, and economies of scale in large multiplexers. *Queueing Systems*, 20: 293 – 320, 1995.

[3] O. Boxma and V. Dumas. Fluid queues with long-tailed activity period distributions. *Computer Communications*, 21: 1509 – 1529, 1998.

[4] C. Courcoubetis and R. Weber. Buffer overflow asymptotics for a buffer handling many traffic sources. *Journal of Applied Probability*, 33: 886 – 903, 1996.

[5] A. Dembo and O. Zeitouni. *Large Deviations Techniques and Applications.* Jones and Bartlett, Boston, 1993.

[6] N. Duffield. Queueing at large resources driven by long-tailed M/G/∞-modulated processes. *Queueing Systems*, 28: 245 – 266, 1998.

[7] V. Dumas and A. Simonian. Asymptotic bounds for the fluid queue fed by sbexponential on-off sources. *Preprint.*

[8] A. Elwalid, D. Mitra, and R. Wentworth. A new approach for allocating buffers and bandwidth to heterogeneous, regulated traffic in an ATM node. *IEEE Journal on Selected Areas in Communications*, 13: 1115 – 1127, 1995.

[9] D. Heyman and T. Lakshman. What are the implications of long-range dependence for VBR traffic engineering? *IEEE/ACM Transactions on Networking*, 4: 301 – 317, 1996.

[10] J. Hui. Resource allocation for broadband networks. *IEEE Journal on Selected Areas in Communications*, 6: 1598 – 1608, 1988.

[11] W. Leland, M. Taqqu, W. Willinger, and D. Wilson. On the self-similar nature of Ethernet traffic. *IEEE/ACM Transactions on Networking*, 2: 1 – 15, 1994.

[12] N. Likhanov and R. Mazumdar. Cell loss asymptotics in buffers fed with a large number of independent stationary sources. *Proceedings IEEE Infocom*, 339 – 346, 1998.

[13] V. Paxson and S. Floyd. Wide area traffic: the failure of Poisson modeling. *IEEE/ACM Transactions on Networking*, 3: 226 – 244, 1995.

[14] B. Ryu and A. Elwalid. The importance of long-range dependence of VBR video traffic in ATM traffic engineering: myths and realities. *Computer Communication Review*, 26: 3 – 14, 1996.

[15] A. Shwartz and A. Weiss. *Large deviations for performance analysis, queues, communication, and computing.* Chapman and Hall, New York, 1995.

[16] A. Simonian and J. Guibert. Large deviations approximation for fluid queues fed by a large number of on/off sources *IEEE Journal of Selected Areas in Communications*, 13: 1017 – 1027, 1995.

[17] A. Weiss. A new technique of analyzing large traffic systems. *Advances of Applied Probability*, 18: 506 – 532, 1986.

OVERFLOW AND LOSSES IN A NETWORK QUEUE WITH SELF-SIMILAR INPUT

Boris Tsybakov,
QUALCOMM Inc.
5775 Morehouse Drive
San Diego, CA 92121-1714
USA
e-mail: borist@qualcomm.com

Nicolas D. Georganas,
Multimedia Communications Research
Laboratory (MCRLab)
School of Info. Tech. & Engin. (SITE)
University of Ottawa
Ottawa, Ontario, Canada K1N 6N5
e-mail: georgana@mcrlab.uottawa.ca

Abstract

This paper considers a discrete time queuing system which models a communication network multiplexer which is fed by a self-similar packet traffic. The model has a finite buffer, a number of servers with unit service time, and an input traffic which is an aggregation of independent source-active periods having Pareto-distributed lengths and arriving as Poisson batches. The new asymptotic upper bounds to the buffer-overflow and packet-loss probabilities P are obtained. These bounds decay algebraically slow in buffer-size h and exponentially fast with excess of channel capacity over traffic rate. A comparison of our upper bounds and the known upper and lower bounds is done. The obtained upper bounds give an exact asymptotic of $\log P/\log h$ when $h \to \infty$.

1. INTRODUCTION. Following the recent discovery on the self-similar nature of traffic flowing in high-speed communications networks (Leland, Taqqu, Willinger, and Wilson [9] and [10], Crovella and Bestavros [4]), much attention was paid on the performance analysis of queueing systems fed by a long-range dependent input process (Willinger, Taqqu, and Erramilli [28]). There was a widely-shared feeling, based on the experimental measurements [9], that an important performance measure of buffer overflow, the overflow probability, decays significantly slower with growing buffer size, under self-similar traffic, than under short-range dependent traffic such as the renewal or Markov-type traffics traditionally used in telecommunication models.

In this paper, we consider a discrete-time queue with a buffer of *finite* size h and C servers with service time 1. The queue is fed by a self-similar traffic which is an aggregation of independent source active periods with rates R, heavy-tailed Pareto (with parameter $1<\alpha<2$) distributed lengths, $\tau \in \mathbf{N}$, and independent Poisson (with parameter λ) number of new source arrivals at each time [11], [17], [21] - [24]). Our interest is in the analysis of the buffer-overflow probability, P_{over}, in the queue. We present a new asymptotic (as $h \to \infty$) upper bound to P_{over}

$$P_{\text{over}} \le a_1 h^{(-\alpha+1)k}, \qquad k=\min\{n: R(n+\lambda \mathrm{E}\tau)>C\}, \qquad h\to\infty, \qquad C>\lambda R\mathrm{E}\tau \tag{1.1}$$

where a_i is a factor independent of h.

The bound (1.1) is tighter than our previous bound [21],

$$P_{\text{over}} \le a_2 h^{(-\alpha+1)}, \qquad h\to\infty, \qquad C>\lambda R\mathrm{E}\tau. \tag{1.2}$$

In the case of $R = C = 1$, the bounds (1.1) and (1.2) have the same exponent $-\alpha+1$ of h although the factors a_1 and a_2 are different and there are the regions of parameters λ, α, R, C where one factor is tighter than another. When $R = C = 1$, the lower bounds to P_{over}, which we proved in [17] and [23],

$$P_{\text{over}} \geq a_3 h^{(-\alpha+1)}, \qquad h \to \infty, \tag{1.3}$$

have the same exponent $-\alpha+1$ of h as the upper bounds (1.1) and (1.3) showing the true asymptotic behavior of the buffer-overflow probability [21].

In the general case of R and C, the negative exponent of h in our new upper bound (1.1) is less or equal (this means that (1.1) is tighter) than the correspondent exponent in (1.2).

How is the upper bound (1.1) proved? First, we split the considered queue into two specific queues, Q_1 and Q_2. The input traffic in queue Q_1 (Q_2) consists of only those source active periods of the original traffic, which have the lengths bounded from below (above). Then, we allow the length boundary to converge to infinity. Using the Chernoff bound, we prove that, in this convergence, the queue Q_2 getting in the limit the whole original traffic gives a zero contribution to P_{over} main term, whereas the queue Q_1 getting in the limit a zero intensity traffic gives a contribution which, in fact, forms P_{over}. In other words, we succeeded to extract a main element of the original self-similar traffic, which composes the asymptotic behavior of P_{over}.

Also, in this paper, we get an asymptotic upper bound to P_{loss}, the packet-loss probability. The upper bound has the same form as (1.1) and it is given as a consequence of a general relation between P_{over} and P_{loss}, which is presented in the paper.

In [24], we obtained the asymptotic lower bounds to P_{over} in general case of R and C. So, in the case of an integer ratio C/R (here, we do not review other cases to avoid notation complication), we proved that

$$P_{\text{over}} \geq a_4 h^{(-\alpha+1)\frac{C}{R}}, \qquad h \to \infty \tag{1.4}$$

where a_4 is given in [24]; and (1.4) is proved without the restriction $C > \lambda R \mathrm{E}\tau$. The bound to P_{loss} is the same as (1.4) but with a different factor a_5. With application of (1.4), it was proved recently that the exponent in (1.1) shows the true asymptotic behavior of P_{over} for finite buffer queue [26], [27].

There are other papers, which consider a problem of asymptotic evaluation of buffer-occupancy distribution function, $F(x)$, in the queue considered in this paper but with *infinite* buffer size and $R = 1$ (Parulekar and Makowski [13] - [15], Duffield [6], [7], Duffield and O'Connell [8]). Those papers use a Large Deviation Principle (see Dembo and Zeitouni [5]) and the Gartner-Ellis theorem [5] which allows to apply the Principle to the considered problem. An upper bound to $1 - F(x)$ can be interpreted as an upper bound to P_{over} with $h = x$ [21], [23]. Taking into account this interpretation, we give a review of results of those papers.

By refining the Duffield-O'Connell theorem [8] and by using the Parulekar-Makowski results [14], Duffield [6] obtained the following large deviation upper bound:

$$\limsup_{x\to\infty} \frac{\log(1 - F(x))}{\log x} \leq 1 - (\alpha - 1)(C - \lambda \mathrm{E}\tau), \quad C > \lambda \mathrm{E}\tau. \tag{1.5}$$

Since $(-\alpha+1)(1+\lfloor C-\lambda \mathrm{E}\tau \rfloor) < 1-(\alpha-1)(C-\lambda \mathrm{E}\tau)$ where $\lfloor x \rfloor$ is the integer part of x, the new bound (1.1) is tighter than (1.5) for any λ, α, C from their set of values. In the case of $C=1$, the bound (1.5) does not work in the sense that it is not better than the trivial bound $P_{\mathrm{over}} \le 1$.

Liu, Nain, Towsley, and Zhang [12] proposed an alternative to the approach based on the Gartner-Ellis theorem, that yields the asymptotic lower and upper bounds to $1-F(x)$. They derived the large deviation upper bound,

$$\limsup_{x\to\infty} \frac{\log(1-F(x))}{\log x} \le -\alpha+1, \qquad C>\lambda \mathrm{E}\tau. \tag{1.6}$$

This bound has the same exponent of h as the bound (1.2) but does not reveal any factor, which accompanies $h^{-\alpha+1}$.

As to a lower bound to $1-F(x)$, which is also considered in those papers, it has no interpretation in terms of finite buffer queue, at least for now. It may not necessarily be a lower or an upper bound to P_{over}. Nevertheless, it is interesting to note that, in [12], it is proved that

$$\liminf_{x\to\infty} \frac{\log(1-F(x))}{\log x} \ge (-\alpha+1)(1+\lfloor C-\lambda \mathrm{E}\tau \rfloor), \qquad C>\lambda \mathrm{E}\tau \tag{1.7}$$

which has the same exponent of x as in (1.1) for $R=1$. Thus in the case of $R=1$, we prove that the expression $(-\alpha+1)(1+\lfloor C-\lambda \mathrm{E}\tau \rfloor)$ is the exponent in an upper bound obtained for the *finite*-buffer queue whereas Liu, Nain, Towsley, and Zhang [12] proved that the same expression is the exponent in a lower bound obtained for the *infinite*-buffer queue. It is likely that this exponent gives the true asymptotic behavior for $1-F(x)$ as it gives for P_{over}. In a more general consideration, this exponent, in our opinion, should occur in a large deviation theorem for a sum of certain long-range dependent random variables.

At the end of our brief review, we would like to note that there exists a great literature on the performance analysis of traditional queueing theory models with finite buffer, such as, for example, **M/M/1/h** and **Geo$^{[X]}$/D/1/h** (see, for example, Takagi [16], Besdikian [1], Tsybakov and Papantoni-Kazakos [25]). However, the results of this analysis do not cover the queues with self-similar or long-range dependent input traffic.

Our paper is organized as follows. In Section 2, we present the considered queue model and give the theorem stating the upper bound (1.1) to the overflow probability and its extension to the loss probability. Also, in Subsection 2.7, a relation between P_{over} and P_{loss} is given. Section 3 gives the steps of theorem proof.

2. QUEUE MODEL AND OVERFLOW AND LOSS PROBABILITIES UPPER BOUNDS.

2. 1. Communication system and its queueing model. Consider a discrete-time, $t \in \{\ldots,-1,0,1,\ldots\} \triangleq \mathbf{Z}$, communication system consisting of a finite buffer and a channel. An input traffic $Y=(\ldots,Y_{-1},Y_0,Y_1,\ldots)$, where $Y_t \in \{0,1,\ldots\} \triangleq \mathbf{Z}_+$ is the number of packets arrived at time $t \in \mathbf{Z}$, feeds the buffer. The buffer has a *finite* size h. This means that it can keep not more than h packets at a time. In any slot t, which is the interval $[t,t+1)$ containing only one time moment t from the discrete time axis $\mathbf{Z}$, the channel can transmit (serve) not more than C packets. C is a finite positive integer, $C \in \{1,2,\ldots\} \triangleq \mathbf{N}$, and is called the channel capacity.

The communication system is considered as a queueing one. It is denoted as **Y/D/C/h/d** where **Y** denotes the input traffic Y, **D** stands for the deterministic service time equal 1, C is the number of servers, **h** means that the buffer size is h, and **d** indicates that we take into account a discipline d in the system.

2. 2. Discipline. The most important class of disciplines for this paper is denoted by $D_C(h)$. A discipline d is in $D_C(h)$ if it satisfies the following conditions [25], [20]: (i) If $Y_t + Z_t > 0$ (where Y_t is the number of new packets arrived at t and Z_t is the number of packets which already were in the buffer at t), then $\min\{Y_t + Z_t, C\}$ packets go into service at t. (ii) If $Y_t + Z_t \le h + C$, then no packets are discarded at t. If $Y_t + Z_t > h + C$, instead, then $Y_t + Z_t - h - C$ packets are discarded at t. Which packets are discarded and which packets go into service depend on a specific discipline $d \in D_C(h)$.

2. 3. Input traffic [11], [17], [22]. The considered traffic Y is assumed to be a stream of packers. The packets have equal lengths 1. The packets are assigned to sources, so the traffic is an aggregation of packets generated by sources. The sources are enumerated by $s \in \mathbf{Z}$. A source s starts to generate its packets at time denoted by ω_s ($\omega_s \le \omega_{s+1}$). The moment ω_s is called the time of source s arrival. The source s generates R packers at each time $\omega_s + i - 1$ in time interval $\omega_s, ..., \omega_s + \tau_s - 1$, $i \in \{1, ..., \tau_s\}$. R is called the source rate and is a finite positive integer, $R \in \mathbf{N}$. The time interval $\omega_s, ..., \omega_s + \tau_s - 1$ is called the *active period* of source s, $\tau_s \in \mathbf{N}$ is called the length of the source s active period. Before time ω_s and after time $\omega_s + \tau_s - 1$, the source s does not generate any packets. At any time moment $t \in \mathbf{Z}$, more than one source arrival can occur. By ξ_t, we denote the number of sources arrived at t, that is, $\xi_t \in \mathbf{Z}_+$ is the number of sources started their active periods at t.

Thus,

$$Y_t = \sum_{s \in \mathbf{Z}} \theta_s(t - \omega_s + 1), \qquad t \in \mathbf{Z}, \tag{2.1}$$

where $\theta_s(i) = R$ for $i \in \{1, ..., \tau_s\}$ and $\theta_s(i) = 0$ for $i \le 0$ and $i \ge \tau_s + 1$. This means that Y_t is a total number of packets generated by all active sources at time t.

It is assumed that $\tau_s, s \in \mathbf{Z}$ are i.i.d. for different s ; the numbers of source arrivals, $\xi_t \in \mathbf{Z}_+$, $t \in \mathbf{Z}$ are i.i.d. with $0 < \lambda \triangleq \mathrm{E}\xi_t < \infty$ and $\Pr\{\xi_t = 0\} < 1$; the random variables τ_s are mutually independent of ξ_t and ω_s. Let τ (let ξ) be a generic symbol for τ_s (for ξ_t).

The most important case of traffic Y is the one that has Pareto-type distributed τ and Poissonian ξ,

$$\Pr\{\tau = l\} = c_0 l^{-\alpha-1}, \quad c_0 \triangleq \Big(\sum_{l=1}^{\infty} l^{-\alpha-1}\Big)^{-1}, \quad 1 < \alpha < 2, \quad l \in \mathbf{N}, \tag{2.2}$$

$$\Pr\{\xi = u\} = e^{-\lambda} \lambda^u / u!, \qquad 0 < \lambda < \infty, \qquad u \in \mathbf{Z}_+. \tag{2.3}$$

Such traffic Y is a stationary (in narrow sense), ergodic, and self-similar process [3], [22].

2. 4. Overflow probability. Here, the overflow probability, P_{over}, is defined for the **Y/D/C/h/d** queue.

The event $\{Y_t + Z_t - h - C > 0\}$ in the **Y/D/C/h/d** queue is called the *buffer overflow* at time t. A time t is called the *overflow moment* if at least one packet is discarded at t. We denote the indicator of the overflow moment by A_t, that is, $A_t = 1$ if t is an overflow moment and $A_t = 0$ otherwise.

The *overflow probability* for **Y/D/C/h/d** queue is defined as

$$P_{\text{over}} \triangleq \limsup_{t\to\infty} \Pr\{A_t = 1\}. \tag{2.4}$$

This means that P_{over} is a steady-state probability of buffer overflow if the limit of $\Pr\{A_t = 1\}$ as $t \to \infty$ exists.

2.5. Upper bound to overflow probability. The following theorem states the upper bound (1.1).

Theorem 1. *The overflow probability, P_{over}, in $\mathbf{Y/D/C/h/d}$ queue with the Pareto-type τ, Poissonian ξ, $d \in D_C(h)$, $R = 1$, and $C > \lambda \mathrm{E}\tau$ is upperbounded as*

$$P_{\text{over}} \underset{\sim}{<} \frac{(c_0\lambda(\alpha-1)^{-\alpha}(C+2)^{\alpha-1})^k}{k!} h^{(-\alpha+1)k}, \quad h \to \infty, \quad k = 1 + \lfloor C - \lambda \mathrm{E}\tau \rfloor \tag{2.5}$$

where $g(x) \underset{\sim}{<} f(x), \quad x \to \infty$ *means* $\limsup_{x\to\infty} g(x)/f(x) \le 1$.

In the case of $R \ge 1, C > \lambda R \mathrm{E}\tau$,

$$P_{\text{over}} \underset{\sim}{<} \frac{(\lambda c_0(\alpha-1)^{-\alpha}((C/R)+2)^{\alpha-1}R^{\alpha-1})^k}{k!} h^{(-\alpha+1)k}, \quad h \to \infty, \quad k = 1 + \left\lfloor \frac{C}{R} - \lambda \mathrm{E}\tau \right\rfloor. \tag{2.6}$$

Proof is given in Section 3.

2.6. Relation between overflow and loss probabilities. In this Subsection, we present without proof the general relations between the buffer-overflow and packet-loss probabilities, P_{over} and P_{loss}, when $G = (\ldots, G_{-1}, G_0, G_1, \ldots)$ and $L = (\ldots, L_{-1}, L_0, L_1, \ldots)$, where G_t is the number of new packets arrived at t with $0 < \mathrm{E}G_t < \infty$ and L_t is the number of packets lost at t in a discrete-time queue **G/D/C/h** with $d \in D_C(h)$, are both stationary and ergodic as it is in the above considered model where $G_t = Y_t$. The relations (inequalities) are used in our proof of Theorem 1 and are useful in their own right. (We note that in general, there are examples of queueing systems having $P_{\text{over}} > P_{\text{loss}}$ and $P_{\text{over}} < P_{\text{loss}}$.)

Since, with probability 1 as $T \to \infty$, the ratio (number of lost packets in interval of length T)/T converges to $\mathrm{E}L_t$ and the ratio (number of packets arrived in interval of length T)/T converges to $\mathrm{E}G_t$ by the Birkhoff theorem, we have

$$P_{\text{loss}} \triangleq \lim_{T\to\infty} \frac{\text{number of lost packets in interval of length } T}{\text{number of packets arrived in interval of length } T} = \frac{\mathrm{E}L_t}{\mathrm{E}G_t}$$

where the limit is with probability 1.

The first relation linking P_{over} and P_{loss} is the following:

$$P_{\text{loss}} \ge P_{\text{over}} / \mathrm{E}G_t. \tag{2.7}$$

For the case of the traffic Y defined in Subsection 2.3 with $R = C = 1$, the Pareto-type τ, and the Poisson ξ, (2.7) was given in [23].

The second relation is for the case of the traffic Y with the Poisson ξ, R=1, and $E\tau < \infty$ (but not necessarily with the Pareto-type τ). This is

$$P_{\text{loss}} \le \frac{a(\lambda, C) P_{\text{over}}}{\lambda E\tau}, \qquad a(\lambda, C) \triangleq e^{2\lambda} + e^{(\lambda E\tau)-\lambda} \frac{[(\lambda E\tau) - \lambda]^{1+C}}{(1+C)!}. \tag{2.8}$$

For the considered traffic Y, the inequalities (2.7) and (2.8) show, in particular, that

$$\lim_{\lambda \to 0} \frac{(\lambda E\tau) P_{\text{loss}}}{P_{\text{over}}} = 1. \tag{2.9}$$

This means that $P_{\text{over}} \approx (\lambda E\tau) P_{\text{loss}}$ for a low traffic intensity $\lambda E\tau$.

In the case of $C = 1$, (2.8) was given in [23].

At the end, we extend the inequalities (2.7) and (2.8) to **Y/D/C/h** queue with $R \in \mathbf{N}$. The probabilities P_{loss} and P_{over} are considered now as the functions of $C, R, h,$ i.e. $P_{\text{loss}} = P_{\text{loss}}(C,R,h)$, $P_{\text{over}} = P_{\text{over}}(C,R,h)$. We have

$$P_{\text{loss}}(C,R,h) \ge P_{\text{over}}(\lceil C/R \rceil, 1, \lceil h/R \rceil)/(E\xi)(E\tau) \tag{2.10}$$

for ξ and τ which are not necessarily Poissonian and Pareto-type distributed respectively, and

$$P_{\text{loss}}(\lceil C/R \rceil, 1, \lfloor h/R \rfloor) \le \frac{a(\lambda, \lceil C/R \rceil)}{\lambda E\tau} P_{\text{over}}(C,R,h) \tag{2.11}$$

for a Poisson ξ and not necessarily Pareto-type τ. The symbol $\lceil x \rceil$ denotes the minimum integer which is greater than or equal to x.

2.7. Upper bound to loss probability. In the case of the traffic Y with the Pareto-type τ, the Poisson ξ, and when C/R and h/R are integers ($R \ge 1$), the results of Subsection 2.6 and Theorem 1 allow to get the following asymptotic upper bound to P_{loss}:

$$P_{\text{loss}} \lesssim \frac{(\lambda c_0 (\alpha-1)^{-\alpha} ((C/R)+2)^{\alpha-1} R^{\alpha-1})^k a(\lambda, C/R)}{(\lambda E\tau)\ k!} h^{(-\alpha+1)k},$$
$$h \to \infty, \qquad k = 1 + \left\lfloor \frac{C}{R} - \lambda E\tau \right\rfloor, \qquad C > \lambda R E\tau \tag{2.12}$$

For P_{loss}, the bound (2.12) shows the same asymptotic behavior, $a_5 h^{(-\alpha+1)k}$, as the bound (1.1) shows for P_{over}.

3. PROOF OF THEOREM 1.

Here we present the main steps of Theorem proof in the case of $R = 1$ and under the additional restriction that $C > 1 + \lambda E\tau$. The full proof is submitted for a journal publication.

The proof is based on the three lemmas which are given below without proofs.

Let us consider the **Y/D/C/h/d**, $d \in D_C(h)$ queue (introduced in Section 2) with R=1, $C \in \mathbf{N}$, the Poisson ξ, and the Pareto-type τ. For a given $0 \le \gamma \le 1$, we split the **Y/D/C/h/d** queue into two queues $\mathbf{Y}^{(i)}/\mathbf{D}/\mathbf{C}^{(i)}/\mathbf{h}^{(i)}/\mathbf{d}^{(i)}$, $d(Q_i) \in D_{C^{(i)}}(h^{(i)})$, i=1, 2 denoted as Q_1 and Q_2 respectively, where $d(Q_i)$ is a discipline in Q_i, $Y^{(i)} = (..., Y_{-1}^{(i)}, Y_0^{(i)}, Y_1^{(i)} ...)$,

$$Y_t^{(1)} \triangleq \sum_{s:\tau_s > \gamma h, s\in \mathbf{Z}} \theta_s(t-\omega_s+1), \qquad Y_t^{(2)} \triangleq \sum_{s:\tau_s \le \gamma h, s\in \mathbf{Z}} \theta_s(t-\omega_s+1), \tag{3.1}$$

$$\begin{gathered} Y_t^{(1)} + Y_t^{(2)} = Y_t, \quad C^{(1)} + C^{(2)} = C, \quad h^{(1)} = 0, \quad h^{(2)} = h; \\ C^{(i)}, C \in \mathbf{N}; \quad Y_t^{(i)}, Y_t \in \mathbf{Z}_+, \qquad i = 1, 2. \end{gathered} \tag{3.2}$$

Thus, the input traffic $Y^{(1)} = (..., Y_{-1}^{(1)}, Y_0^{(1)}, Y_1^{(1)}, ...)$ in Q_1 is composed of the traffic Y sources which have the long active periods (with lengths which are greater than γh), and the input traffic $Y^{(2)} = (..., Y_{-1}^{(2)}, Y_0^{(2)}, Y_1^{(2)}, ...)$ in Q_2 is composed of the traffic Y sources which have bounded active periods (with lengths which are not greater than γh). The Q_i-queue has $C^{(i)}$ servers. The Q_1-queue has a zero-size buffer, $h^{(1)} = 0$ (in other words, Q_1 has no buffer). The Q_2-queue has a buffer of size $h^{(2)} = h$; this is the size of buffer in the initial **Y/D/C/h/d** queue also.

We note that the numbers of new sources which come at t in $Y^{(1)}$ and $Y^{(2)}$ are the Poisson random variables with parameters $\lambda_1 \triangleq \lambda \Pr\{\tau > \gamma h\}$ for $Y^{(1)}$ and $\lambda_2 \triangleq \lambda \Pr\{\tau \le \gamma h\}$ for $Y^{(2)}$. The probability distributions $\Pr\{Y_t = n\}$, $\Pr\{Y_t^{(1)} = n\}$, and $\Pr\{Y_t^{(2)} = n\}$, $n \in \mathbf{Z}_+$ are, also, Poissonian with parameters denoted as $\mu_0 \triangleq \mathrm{E}Y_t$, $\mu_1 \triangleq \mathrm{E}Y_t^{(1)}$, and $\mu_2 \triangleq \mathrm{E}Y_t^{(2)}$ respectively. All traffics, Y, $Y^{(1)}$, and $Y^{(2)}$ are stationary and ergodic.

We denote the overflow probability by P_{over} in the **Y/D/C/h/d** queue and by $P_{\text{over}}(Q_i)$ in Q_i. We note that P_{over}, $P_{\text{over}}(Q_1)$, and $P_{\text{over}}(Q_2)$ do not depend on the disciplines in their queues since $d \in D_C(h)$ and $d(Q_i) \in D_{C^{(i)}}(h^{(i)})$ [17].

The following Lemma gives a relation between P_{over}, $P_{\text{over}}(Q_1)$, and $P_{\text{over}}(Q_2)$.

Lemma 1.

$$P_{\text{over}} \le a(\lambda_1, C^{(1)}) P_{\text{over}}(Q_1) + a(\lambda_2, C^{(2)}) P_{\text{over}}(Q_2) \tag{3.3}$$

where $\lambda_1 \triangleq \lambda \Pr\{\tau > \gamma h\}$, $\lambda_2 \triangleq \lambda \Pr\{\tau \le \gamma h\}$ *and* $a(\lambda, C)$ *is defined in* (2.8).

Next, we choose

$$C^{(1)} \triangleq C - C^{(2)} = \lfloor C - \varepsilon - \lambda \mathrm{E}\tau \rfloor \ge 1, \qquad C^{(2)} = \lceil \varepsilon + \lambda \mathrm{E}\tau \rceil \tag{3.4}$$

where $\varepsilon \ge 0$. The condition $C^{(1)} \ge 1$ holds if $C > 1 + \lambda \mathrm{E}\tau$ and ε is sufficiently small.

We want to obtain the upper bounds to $P_{\text{over}}(Q_1)$ and $P_{\text{over}}(Q_2)$ when $C^{(1)}$ and $C^{(2)}$ are chosen as in (3.4).

Lemma 2.

$$P_{\text{over}}(Q_1) \le \frac{(\lambda c_0 (\alpha-1)^{-1} \gamma^{-\alpha+1})^{1+C^{(1)}}}{(1+C^{(1)})!} h^{(-\alpha+1)(1+C^{(1)})} \equiv c_1 h^{(-\alpha+1)(1+C^{(1)})} \tag{3.5}$$

where c_0 *is defined in* (2.2).

The following Lemma will allow us to show that the term with $P_{\text{over}}(Q_2)$ is less enough than the term with $P_{\text{over}}(Q_1)$ in (3.3) as $h \to \infty$. We present the Lemma in the case of $R \geq 1$ although it will be used only for R=1.

Lemma 3. *If* γ *and* ν *are such that*

$$0 < \gamma \leq \frac{\alpha - 1}{(C+2)R} - \nu, \qquad 0 < \nu < \frac{\alpha - 1}{(C+2)R}, \tag{3.6}$$

then

$$P_{\text{over}}(Q_2) \leq \frac{e^{\phi}}{(C+1)\varepsilon} h^{-(1+C-(C+2)\gamma C^{(2)})} \tag{3.7}$$

for any $\phi > 0$, $C^{(2)}$ *given by* (3.4), *the same* $\varepsilon > 0$ *as in* (3.4), *and a sufficiently large* h.

The statement (2.5) under the restriction $C > 1 + \lambda \mathrm{E}\tau$ follows, now, from Lemmas 1, 2, and 3 if we take $\gamma = \dfrac{\alpha - 1}{C+2} - \nu$ and $\varepsilon = \varepsilon(h)$ such that $\varepsilon(h) \to 0$, $h \to \infty$ and $\varepsilon^{-1} h^{-(1+C-(C+2)\gamma C^{(2)})-(-\alpha+1)(1+C^{(1)})} \to 0$, $h \to \infty$ where $C^{(1)}$ and $C^{(2)}$ are given by (3.4).

References

[1] C. Bisdikian," Performance Analysis of Single-Stage, Output Buffer Packet Switches with Independent Batch Arrivals", *Computer Networks and ISDN Systems,* Vol.27, pp.627-652, 1995.

[2] A. A. Borovkov, "*Asymptotic Methods in Queueing Theory*", Wiley, 1984.

[3] D. R. Cox, "Long-Range Dependence: A Review", in *Statistics:An Appraisal,* H. A. David and H. T. David, eds. Ames, IA:The Iowa State University Press, pp. 55-74, 1984.

[4] M. E. Crovella and A. Bestavros, "Self-Similarity in World Wide Web Traffic: Evidence and possible causes", *Proceedings of the 1996 ACM SIGMETRICS. International Conference on Measurement and Modeling of Computer Systems,* May, 1996 and *IEEE/ACM Trans. on Networking,* Vol. 5, No. 6, pp. 835-846, 1997.

[5] A. Dembo and O. Zeitouni," *Large Deviation Techniques and Applications",* Jones and Bartlett, Boston (MA), 1993.

[6] N. G. Duffield, "On the Relevance of Long-Tailed Durations for the Statistical Multiplexing of Large Aggregations", *Proc. of the 34-th Annual Allerton Conf. on Communication, Control, and Computing,* Oct. 2-4, pp. 741-750, 1996.

[7] N. G. Duffield, "Queueing at Large Resources Driven by Long-Tailed $M/G/\infty$-modulated Processes", a manuscript, December 30, 1996.

[8] N. G. Duffield and N. O'Connell, "Large Deviations and Overflow Probabilities for the General Single-server Queue with Applications", *Math. Proc. Cam. Phil. Soc.,* Vol.118, pp.363-374, 1995.

[9] W. E. Leland, M. S. Taqqu, W. Willinger and D. V. Wilson, "On the Self-Similar Nature of Ethernet Traffic", *Proc. ACM SIGCOMM'93,* San Fransisco, CA, pp. 183-193, 1993.

[10] W. E. Leland, M. S. Taqqu, W. Willinger, and D. V. Wilson, "On the Self-Similar Nature of Ethernet Traffic (Extended version)", *IEEE/ACM Trans. on Networking,* Vol. 2, No. 1, pp. 1-15, 1994.

[11] N. Likhanov, B. Tsybakov, and N. D. Georganas, "Analysis of an ATM Buffer with Self-Similar ("Fractal") Input Traffic", *Proc. IEEE INFOCOM'95*, Boston, MA, 1995, pp. 985-992.

[12] Z. Liu, P. Nain, D. Towsley, and Z. -L. Zhang, "Asymptotic Behavior of a Multiplexer Fed by a Long-Range Dependent Process", *CMPSCI Technical Report 97-16,* University of Massachusetts at Amherst, 1997.

[13] M. Parulekar and A. M. Makowski, "Tail Probabilities for a Multiplexer with Self-Similar Traffic", *Proc. of the IEEE INFOCOM'96 Conf.,* Mar. 26-28, 1996, pp. 1452-1459, 1996.

[14] M. Parulekar and A. M. Makowski, "Tail Probabilities for $M/G/\infty$ Input Processes (I): Preliminary Asymptotics", Preprint, University of Maryland, 1996.

[15] M. Parulekar and A. M. Makowski, "$M/G/\infty$ Input Processes: A Versatile Class of Models for Network Traffic", Preprint, University of Maryland, 1996.

[16] H. Takagi,*"Queueing Analysis; A Foundation of Performance Evaluation",* Vol. I - III, Elsevier Science Publishers B.V. (North-Holland) Amsterdam, 1991.

[17] B. Tsybakov and N. D. Georganas, "On Self-Similar Traffic in ATM Queues: Definitions, Overflow Probability Bound and Cell Delay Distribution", *IEEE/ACM Trans. on Networking,* Vol. 5, No. 3, pp. 397-409, 1997.

[18] B. Tsybakov and N. D. Georganas, "Self-Similar Traffic: Upper Bounds to Buffer-Overflow Probability in an ATM Queue", *Proc. CCBR'97, the Canadian Conference on Broadband Research, Ottawa,* April 1997, pp. 137-148, 1997.

[19] B. Tsybakov and N. D. Georganas, "Overflow Probability in an ATM Queue with Self-Similar Traffic", *IEEE ICC'97, Montreal, Conference Record,* Vol. 2, pp. 822-826, 1997.

[20] B. Tsybakov and N. D. Georganas, "A Model of Self-Similar Communications-Network Traffic", *Proc. of International Conf. "Distributed Computer Communication Networks" (DCCN'97),* Tel-Aviv, Israel, pp. 212-217, 1997.

[21] B. Tsybakov and N. D. Georganas, "Self-Similar Traffic and Upper Bounds to Buffer-Overflow Probability in ATM Queue", *Performance Evaluation.* , Vol. 32, pp. 57-80, 1998.

[22] B. Tsybakov and N. D. Georganas, "Self-Similar Processes in Communications Networks", *IEEE Trans. on Information Theory*, Vol.44, No.5, pp.1713-1725, 1998.

[23] B. Tsybakov and N. D. Georganas, "Overflow and Loss Probabilities in a Finite ATM Buffer Fed by Self-Similar Traffic", *Queueing Systems* (to appear in 1999).

[24] B. Tsybakov and N. D. Georganas, "On Buffer Overflow under Self-Similar Traffic", *Proc. of SPIE*, Vol. 3841, 1999.

[25] B. Tsybakov and P. Papantoni-Kazakos, "The Best and Worst Packet Transmission Policies", *Problems of Information Transmission,* Vol.32, No. 4, pp. 365-382, 1996.

[26] B. Tsybakov, "Communication Networks with Self-Similar Traffic", Symposium *"Numbers, Information Theory and Complexity",* in honour of R. Ahlswede, Eds. B. Balkenhol and U. Tamm, Erganzungsreihe 98-010, Universitat Bielefeld (Germany), 1998.

[27] B. Tsybakov, "A Model of Teletraffic on the Basis of a Self-Similar Signal Process", *Radiotechnika,* No.5, pp. 24-31, 1999 (in Russian). English translation in *"Radio and Communication Technology"* by Allerton Press Inc.

[28] W. Willinger, M. S. Taqqu, and A. Erramilli, "A Bibliographical Guide to Self-Similar Traffic and Performance Modeling for Modern High-Speed Networks" in *"Stochastic networks:Theory and applications",* Ed. F. P. Kelly, S. Zachary and I. Ziedins, Clarendon Press (Oxford University Press), Oxford, pp. 339-366, 1996.

CHOKe
A stateless mechanism for providing Quality of Service in the Internet

Rong Pan, Balaji Prabhakar
Department of Electrical Engineering
Stanford University
Stanford, CA 94305
{rong, balaji}@leland.stanford.edu

Abstract

We investigate the problem of providing a fair bandwidth allocation to each of n flows that share the outgoing link of a congested router. The buffer at the outgoing link is a simple FIFO, shared by packets belonging to the n flows. We devise a simple packet dropping scheme, called CHOKe, that discriminates against the flows which submit more packets/sec than is allowed by their fair share. By doing this, the scheme aims to approximate the fair queueing policy. Since it is stateless and easy to implement, CHOKe controls unresponsive or misbehaving flows with a minimum overhead.

Keywords

fair queueing, RED, active queue management, scheduling algorithm.

I. INTRODUCTION

The Internet provides a connectionless, best effort, end-to-end packet service using the IP protocol. It depends on congestion avoidance mechanisms implemented in the transport layer protocols, like TCP, to provide good service under heavy load. However, a lot of TCP implementations do not include the congestion avoidance mechanism either deliberately or by accident. Moreover, there are a growing number of UDP-based applications running in the Internet, such as packet voice and packet video. The flows of these applications do not back off properly when they receive congestion indications. As a result, they aggressively use up more bandwidth than other TCP compatible flows. This could eventually cause "Internet Meltdown" [2]. Therefore, it is necessary to have router mechanisms to shield responsive flows from unresponsive or aggressive flows and to provide a good quality of service (QoS) to all users.

As discussed in [2], there are two types of router algorithms for achieving congestion control, broadly classified under the monikers "scheduling algorithms" and "queue management algorithms". The generic scheduling algorithm, exemplified by the well-known Fair Queueing (FQ) algorithm, requires the buffer at each output of a router to be partitioned into separate queues each of which will buffer the packets of one of the flows [1][3]. Packets from the flow buffers are placed on the outgoing line by a scheduler using an approximate bit-by-bit, round-robin discipline. Because of per flow queueing, packets belonging to different flows are essentially isolated from each other and one flow cannot degrade the quality of another. However, it is well-known that this approach requires complicated per flow state information, making it too expensive to be widely deployed.

To reduce the cost of maintaining flow state information, Stoica et al [17] have recently proposed a scheduling algorithm called Core Stateless Fair Queueing (CSFQ). In this method routers are divided into two categories: edge routers and core routers. An edge router keeps per flow state information and estimates each flow's arrival rate. These estimates are inserted into the packet headers and passed on to the core routers. A core router simply maintains a stateless FIFO queue and, during periods of congestion, drops a packet randomly based on the rate estimates. This scheme reduces the core router's design complexity. However, the edge router's design is still complicated. Also, because of the rate information in the header, the core routers have to extract packet information differently from traditional routers. Another notable scheme which aims to approximate FQ at a smaller implementation cost is Stochastic Fair Queueing (SFQ) proposed by McKenny [13]. SFQ classifies packets into a smaller number of queues than FQ using a hash function. Although this reduces FQ's design complexity, SFQ still requires around 1000 to 2000 queues in a typical router to approach FQ's performance [12].

Thus, scheduling algorithms can provide a fair bandwidth allocation, but they are often too complex for high-speed implementations and do not scale well to a large number of users. On the other hand, queue management algorithms have had a simple design from the outset. Given their simplicity, the hope is to approximate fairness. This class of algorithms is exemplified by Random Early Detection (RED) [5]. A router implementing RED maintains a single FIFO to be shared by all the flows, and drops an arriving packet at random during periods of congestion. The drop probability increases with the level of congestion. Since RED acts in *anticipation* of congestion, it does not suffer from the "lock out" and "full queue" problems [2] inherent in the widely deployed Drop Tail mechanism. By keeping the average queue-size small, RED reduces the delays experienced by most flows. However, like Drop Tail, RED is unable to penalize unresponsive flows. This is because the percentage of packets dropped from each flow over a period of time is almost the same. Consequently, misbehaving traffic can take up a large percentage of the link bandwidth and starve out TCP friendly flows.

To improve RED's ability for distinguishing unresponsive users, a few variants (like RED with penalty box [6] and Flow Random Early Drop (FRED) [11]) have been proposed. However, these variants incur extra implementation overhead since they need to collect certain types of state information. RED with penalty box stores information about unfriendly flows while FRED needs information about active connections. The recent paper by Ott et al [14] proposes an interesting algorithm called Stabilized RED (SRED) which stabilizes the occupancy of the FIFO buffer, independently of the number of active flows. More interestingly, SRED estimates the number of active connections and finds candidates for misbehaving flows. It does this by maintaining a data structure, called the "Zombie list", which serves as a proxy for information about recently seen flows. Although SRED identifies misbehaving flows, it does not propose a simple router mechanism for penalizing misbehaving flows. The CHOKe algorithm proposed below simultaneously identifies and penalizes misbehaving flows, and is simpler to implement than SRED.

In summary, all of the router algorithms (scheduling and queue management) developed thus far have been either able to provide fairness or simple to implement, but not both simultaneously. This has led to the belief that the two goals are somewhat incompatible (see [18]).

This paper takes a step in the direction of bridging fairness and simplicity. Specifically, we exhibit an active queue management algorithm, called CHOKe, that is simple to implement (since it requires no state information) and differentially penalizes misbehaving flows by dropping more of their packets. By doing this, CHOKe (CHOose and Keep for responsive flows, CHOose and Kill for unresponsive flows) aims to approximate max-min fairness for the flows that pass through a congested router.

The basic idea behind CHOKe is that the contents of the FIFO buffer form a "sufficient statistic" about the incoming traffic and can be used in a simple fashion to penalize misbehaving flows. When a packet arrives at a congested router, CHOKe draws a packet at random from the FIFO buffer and compares it with the arriving packet. If they both belong to the same flow, then they are both dropped, else the randomly chosen packet is left intact and the arriving packet is admitted into the buffer with a probability that depends on the level of congestion (this probability is computed exactly as in RED). The reason for doing this is that the FIFO buffer is more likely to have packets belonging to a misbehaving flow and hence these packets are more likely to be chosen for comparison. Further, packets belonging to a misbehaving flow arrive more numerously and are more likely to trigger comparisons. The intersection of these two high probability events is precisely the event that packets belonging to misbehaving flows are dropped. Therefore, packets of misbehaving flows are dropped more often than packets of well-behaved flows[1].

The rest of the paper is organized as follows: Section II explains our motivation and goals for using the CHOKe mechanism and describes the CHOKe algorithm (and a few variants) in detail. The simulation results are presented in Section III. Section IV summaries the analytical models briefly. Our conclusions are presented in Section V.

II. MOTIVATION, GOALS, AND THE ALGORITHM

Our work is motivated by the need for a simple, stateless algorithm that can achieve flow isolation and/or approximate fair bandwidth allocation. As mentioned in the introduction, existing algorithms (like RED, FQ and others) are either simple to implement or able to achieve flow isolation, but not both simultaneously.

We seek a solution to the above problem in the context of the Internet. Thus, we are motivated to find schemes that differentially penalize "unfriendly" or "unresponsive" flows[2], which implies bad implementations of TCP, and UDP-based flows. Further, we seek to preserve some key features that RED possesses; such as its ability to avoid global synchronization[3], its ability to keep buffer occupancies small and ensure low delays, and its lack of bias against bursty traffic. By doing this CHOKe performs similarly to RED in the absence of unfriendly or unresponsive flows.

Next, we need a benchmark to compare the extent of fairness achieved by our solution. Max-min fairness suggests itself as a natural candidate for two reasons: (a) It is well-defined and widely understood in the context of computer networks (see [1], page 526, or [10]), and (b) the FQ algorithm is known to achieve it. However, for any scheme to achieve perfect max-min fairness without flow state information seems almost impossible. Max-min fairness is not suitable in our context since we do not identify the flow(s) with the minimum resource allocation and maximize its (their) allocation. Instead we identify and reduce the allocation of the flows which consume the most resources. In other words, we attempt to *minimize the resource consumption of the maximum flow(s)* or seek to achieve *min-max fairness*. [16] defines min-max fairness and proves that the min-max allocation vector equals the max-min allocation vector.

---

[1] To our knowledge, the only other algorithm that makes a random comparison to identify misbehaving flows is SRED. The idea of making a random comparison, observed independently by us, is directly taken advantage of in CHOKe (i.e. without maintaining state information) to differentially drop packets belonging to misbehaving flows.

[2] See [7] for a formal definition of these terms.

[3] Global synchronization refers to the situation where a lot of connections decrease or increase their window size at the same time, as happens under the Drop Tail mechanism (cf. [5]).

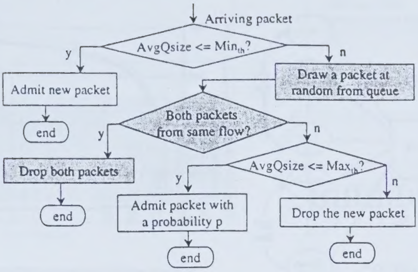


Fig. 1. The CHOKe algorithm

A. The CHOKe algorithm

Suppose that a router maintains a single FIFO buffer for queueing the packets of all the flows that share an outgoing link. We describe an algorithm, CHOKe, that differentially penalizes unresponsive and unfriendly flows. The state, taken to be the number of active flows and the flow ID of each of the packets, is assumed to be unknown to the algorithm. The only observable for the algorithm is the total occupancy of the buffer.

CHOKe calculates the average occupancy of the FIFO buffer using an exponential moving average, just as RED does. It also marks two thresholds on the buffer, a minimum threshold min_{th} and a maximum threshold max_{th}.

If the average queue size is less than min_{th}, every arriving packet is queued into the FIFO buffer. When the average queue size is bigger than min_{th}, each arriving packet is compared with a randomly selected packet, called drop candidate packet, from the FIFO buffer. If they have the same flow ID, they are both dropped. Otherwise, the randomly chosen packet is replaced (in the same position as before) and the arriving packet is dropped with a probability that depends on the average queue size. The drop probability is computed exactly as in RED. In particular, this means that packets are dropped with probability 1 if they arrive when the average queue size exceeds max_{th}. A flow chart of the algorithm is given in Figure 1. In order to bring the queue occupancy back to below max_{th} as fast as possible, we still compare and drop packets from the queue when the queue size is above the max_{th}.

In general, one can choose $m > 1$ packets from the buffer, compare all of them with the incoming packet, and drop the packets that have the same flow ID as the incoming packet. Not surprisingly, we shall find that choosing more than one drop candidate packet improves CHOKe's performance. This is especially true when there are multiple unresponsive flows; indeed, as the number of unresponsive flows increases, it is necessary to choose more drop candidate packets. However, since we insist on a completely stateless design, we cannot a priori know how many unresponsive flows are active at any time (and then choose a suitable value for m). It turns out that we can automate the process so that the algorithm chooses the proper value of $m \geq 1$. One way of achieving this is to partition the interval between min_{th} and max_{th} into k regions $R_1, R_2, \ldots, R_k$ and choose different values of m depending on the region the average buffer occupancy lies in. For example, we could choose $m = 2 \cdot i$ $(i = 1, \ldots, k)$, when the average queue size lies in region R_i. Obviously, we need to let m increase monotonically with the average queue size.

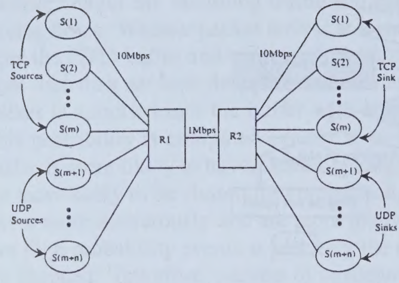


Fig. 2. Network Configuration: m TCP sources, n UDP sources

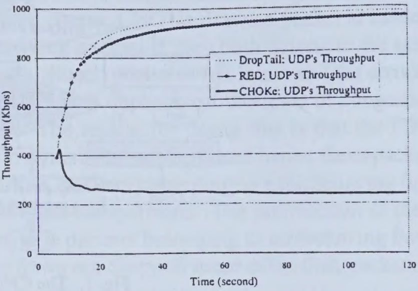


Fig. 3. UDP Throughput Comparison

CHOKe is a stateless algorithm. It does not require any special data structure. Compared to a pure FIFO queue, there are just a few simple extra operations that CHOKe needs to perform: draw a packet randomly from the queue, compare flow IDs, and possibly drop both the incoming and the candidate packets. Since CHOKe is embedded in RED, it inherits the good features of RED mentioned previously. Finally, as a stateless algorithm, it's nearly as simple to implement as RED. To see this, let us consider the details of implementation. Drawing a packet at random can be implemented by generating a random address from which a packet flow ID is read out. Flow ID comparison can be done easily in hardware. It is arguably more difficult to drop a randomly chosen packet since this means removing it from a linked-list. Instead of doing this, we propose to add one extra bit to the packet header. The bit is set to one if the drop candidate is to be dropped. When a packet advances to the head of the FIFO buffer, the status of this bit determines whether it is to be immediately discarded or transmitted on the outgoing line.

III. SIMULATION RESULTS

This section presents simulation results of CHOKe's performance in penalizing misbehaving flows and thus approximating fair bandwidth allocation. We shall use the RED and Drop Tail schemes, whose complexities are close to that of CHOKe, for comparison. The simulations range over a spectrum of network configurations and traffic mixes. The results are presented in three parts: single congested link, multiple congested links, and multiple misbehaving flows.

A. Single Congested Link

To illustrate CHOKe's performance when there is a single congested link, we consider the standard network configuration shown in Figure 2. The congested link in this network is between the routers $R1$ and $R2$. The link, with capacity of 1 Mbps, is shared by m TCP and n UDP flows. An end host is connected to the routers using a 10 Mbps link, which is ten times the bottleneck link bandwidth. All links have a small propagation delay of 1ms so that the delay experienced by a packet is mainly caused by the buffer delay rather than the transmission delay. The maximum window size of TCP is set to 300 (so that it doesn't become a limiting factor of a flow's throughput). The TCP flows are derived from FTP sessions which transmit large sized files. The UDP hosts send packets at a constant bit rate (CBR) of r Kbps, where r is a variable. All packets are set to have a size of 1K Bytes.

To study how much bandwidth a single nonadaptive UDP source can obtain when routers use

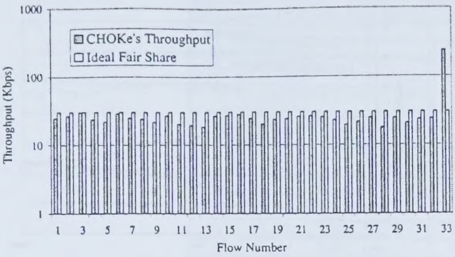


Fig. 4. CHOKe: Throughput Per Flow

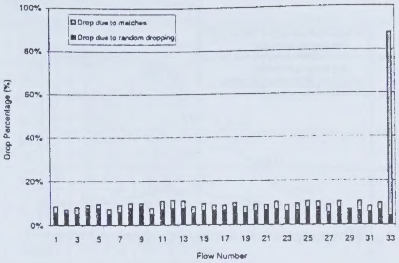


Fig. 5. CHOKe: Drop Decomposition

different queue management schemes, we set up the following simulation: there are 32 TCP sources ($Flow1$ to $Flow32$) and 1 UDP source ($Flow33$) in the network. The UDP source sends packets at a rate r = 2 Mbps, twice the bandwidth of the bottleneck link, so that the link $R1$-$R2$ becomes congested. The minimum threshold min_{th} in the RED and CHOKe schemes is set to 100, allowing on average around 3 packets per flow in the buffer before a router starts dropping packets. Following [5], we set the maximum threshold max_{th} to be twice min_{th}, and the physical queue size is fixed at 300 packets. The throughput of the UDP flow under different router algorithms: DropTail, RED and CHOKe, is plotted in Figure 3.

From Figure 3, we can clearly see that the RED and DropTail gateways do not discriminate against unresponsive flows. The UDP flow takes away more than 95% of the bottleneck link capacity and the TCP connections can only take the remaining 50 Kbps. CHOKe, on the other hand, improves the throughput of the TCP flows dramatically by limiting the UDP throughput to about 250 Kbps, which is only 25% of the link capacity. The total TCP flows' throughput is boosted from 50 Kbps to 750 Kbps.

To gauge the degree to which CHOKe achieves fair bandwidth allocation, the individual throughput of each of the 33 connections in the simulation above, along with their ideal fair shares, are plotted in Figure 4. Although the throughput of the UDP flow ($Flow33$) is still higher than the rest of the TCP flows, it can be seen that each TCP is allocated a bandwidth relatively close to its fair share. In CHOKe, a packet could be dropped because of a match or a random discard like in RED. A misbehaving flow, which has a high arrival rate and a high buffer occupancy, incurs packet dropping mostly due to matches. On the other hand, a match is rarer for the packets of a responsive flow and drops occur mainly due to random discard. In the above simulation we find that matches are responsible for 85% of UDP packet drops, while only 30% of the TCP packet drops are due to matches. The drop decomposition is shown in Figure 5.

Next, we vary the UDP arrival rate r to study CHOKe's performance under different traffic load conditions. The simulation results are summarized in Figure 6, where the UDP's throughput versus the UDP flow arrival rate is plotted. The drop percentage of the UDP flow is also shown in the figure. The values of min_{th} and max_{th} are set at 30 and 60 respectively to study the performance of CHOKe under different threshold settings. From the plot, we see that CHOKe drops 23% of the UDP packets when its arrival rate is as low as 100 Kbps. As the UDP arrival rate increases, the drop percentage goes up as well. The average TCP flow's throughput stays almost constant. For comparison, RED's performance under different traffic load is also shown in Figure 6. As can be seen RED is unable to discriminate against unresponsive flows.

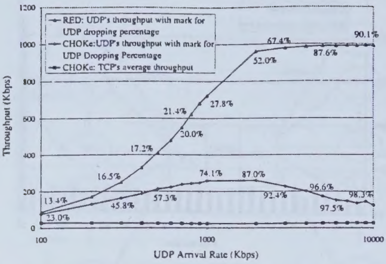


Fig. 6. Performance under Different Traffic Load

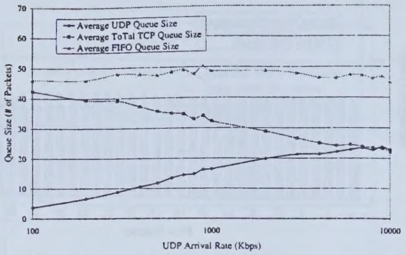


Fig. 7. CHOKe's Queue Distribution Under Different Loadings

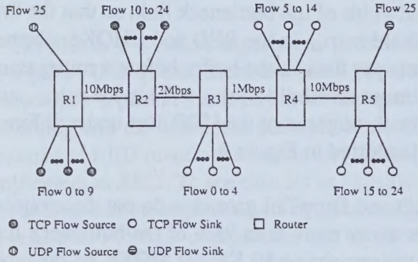


Fig. 8. Topology of Multiple Links

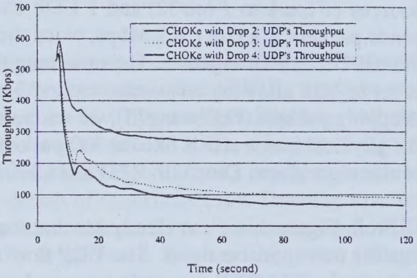


Fig. 9. CHOKe with Multiple Drops: Throughput Comparison

Figure 7 shows the queue distribution among the flows for different traffic load conditions. It is interesting to note that the occupancy of the UDP flow increases monotonically with its offered rate. This increases the chance that a match is obtained for UDP packets. Since two packets are dropped when there is a match, as UDP occupancy approaches 50% (and hence the probability of a match conditioned on a UDP arrival approaches 0.5), each arriving UDP packet causes on average one UDP drop. That is, the proportion of dropped UDP packets approaches 1 and the UDP flow's goodput goes to zero. This intuitively explains why the UDP throughput decreases drastically under heavy loading.

B. Multiple Congested Links

So far we have seen the performance of CHOKe in a simple network configuration with one congested link. In this section, we study how CHOKe performs when there are multiple congested links in the network. A sample network configuration with five routers is constructed in Figure 8. The first link between router *R1* and *R2* (*R1-R2*) has a capacity of 10 Mbps so that when the sources connected to it send packets at a high rate, the following link *R2-R3* becomes congested. The third link, *R3-R4*, has only half the bandwidth of the link *R2-R3*and becomes congested since the link's arrival rate exceeds its capacity. The final link *R4-R5* is lightly loaded. Using these links, we can demonstrate CHOKe's performance under cascaded, multiple congested links. In total there are 25 TCP flows and 1 UDP flow in the network, whose sources and sinks are shown in the figure. The UDP source sends packets at a rate of 2 Mbps while the other network parameters remain the same as in the setting of Figure 6. Table I lists the throughput of each flow at various links in the network. We see that after competing

Flowid	T_{R2-R3}	T_{R3-R4}	T_{R4-R5}	Flowid	T_{R2-R3}	T_{R3-R4}	T_{R4-R5}
0	107.715	-	-	13	33.979	32.488	-
1	117.993	-	-	14	35.010	33.193	-
2	108.149	-	-	15	33.898	32.352	32.352
3	114.630	-	-	16	33.518	31.837	31.837
4	114.277	-	-	17	34.169	32.569	32.569
5	33.654	32.244	-	18	34.820	32.949	32.949
6	34.603	33.328	-	19	34.820	33.437	33.437
7	33.464	31.674	-	20	35.362	34.359	34.359
8	35.715	34.576	-	21	35.823	34.494	34.494
9	33.789	32.569	-	22	34.955	33.138	33.138
10	34.088	32.216	-	23	33.816	32.379	32.379
11	37.396	36.094	-	24	36.013	34.522	34.522
12	37.884	36.637	-	25	740.311	332.854	332.854

TABLE I

EACH FLOW'S THROUGHPUT AT DIFFERENT LINKS (T_{Ri-Rj}=THROUGHPUT AT LINK RI-RJ)

with 25 TCP flows in the first congested link *R2-R3*, the UDP flow loses around 63% of its traffic when it sends data at the same rate as the link bandwidth (2 Mbps). The UDP traffic that gets through to the next link, *R3-R4*, constitutes 74% of the arrival rate at that link. The flow suffers an additional 55% loss at this link when competing with the remaining 20 TCP flows. Comparing these results with the single congested link case, we see that each of the cascaded congested links behaves roughly as if it was a single congested link. Multiple congested links therefore have an additive effect on UDP packet losses. On the other hand, TCP flows can automatically detect their bottleneck link bandwidth and therefore they suffer much less loss.

From the simulation results in Table I and the discussion above, one infers that since TCP flows are responsive to congestion indication and adjust their packet injection rates accordingly, their packet loss rate in a network under the CHOKe scheme is quite small. But nonadaptive flows, like UDP, suffer severe packet losses.

C. Multiple Drop Candidates and Misbehaving Flows

We now study the effect of the generalized CHOKe algorithm where more than one drop candidate is drawn from the buffer. Figure 9 shows the performance of the CHOKe algorithm with one, two and three drop candidates for the network configuration of Figure 2. The rate for the UDP source is 2 Mbps. Since CHOKe with $m-1$ candidates has a maximum drop of m packets ($m-1$ candidate packets $+1$ incoming packet), it will be referred to as CHOKe with drop m. Not surprisingly, Figure 9 shows that CHOKe with multiple drops has a better control over the unresponsive UDP traffic[4].

When there are many UDP flows in the network, CHOKe with multiple drops exhibits its advantage over the basic algorithm (i.e. CHOKe with drop 2). A simulation configuration with 32 TCP and 5 UDP sources is set up, similar to the configuration in Figure 2. All the UDP sources are assumed to have the same arrival rate. The values of min_{th} and max_{th} are still set at 30 and 60 respectively. The results for CHOKe with drop 2 and CHOKe with drop 5 are given in Figure 10. As shown in the figure, the throughput of the UDP sources goes up monotonically with their arrival rate. As a result, there is almost no bandwidth left for the TCP sources. Although the total UDP flows occupy almost all the buffer space, since each UDP connection takes only around 20% of the queue. As a result, the chance of obtaining a match

---

[4] However, it is interesting to observe that the performance improvement from CHOKe drop 3 to CHOKe drop 4 is very small.

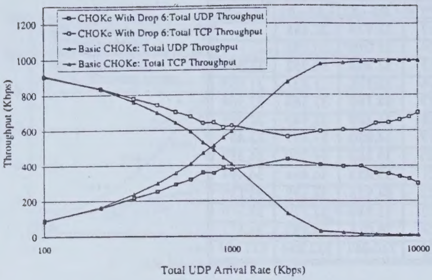


Fig. 10. CHOKe: Throughput for 32 TCP and 5 UDP configuration

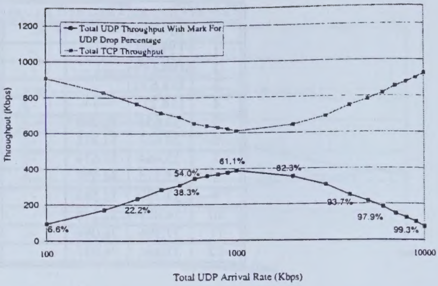


Fig. 11. Self Adjusting CHOKe: Throughput for 32 TCP and 5 UDP configuration

for the packets of a UDP flow are slim.

On the other hand, CHOKe with drop 5 boosts the throughput of the TCP flows. Clearly, choosing multiple drop candidates improves the chance of dropping UDP packets and improving TCP throughput.

Figure 10 demonstrates the need for multiple drops when there are multiple unresponsive flows. However, the number of unresponsive traversing link will typically be unknown to the algorithm. One way of automating the process, which determines the appropriate number of packets to be drawn from the queue, is to observe that the average queue size is a good indicator of the congestion level. For a fixed service rate, congestion increases with the arrival rate and, more importantly, with a decrease in the number of drops. We use this observation and automate as follows: Divide the region between min_{th} and max_{th} into a number, say k, of subregions. If the average queue size lies in region i (where region i is closer to min_{th} than region $i+1$), then the number of drops is set to $f(i)$ $(i = 1 \dots k)$, where $f(i)$ is a monotonically increasing function of i. Figure 11 shows a plot of the performance of the self adjusting mechanism when there are 5 UDPs. In this plot k is chosen to be 8 and $f(i) = 2 \cdot i$.

IV. MODELS

Due to page restrictions this section is very brief, mostly summarizing material presented elsewhere ([16]). Since tractable models for the actual CHOKe algorithm are hard to find, we introduce and study two close variants: Front and Back CHOKe. These differ from the actual algorithm in the sampling distribution; i.e. the location in the FIFO buffer from which candidates are chosen for comparison with the incoming packet. Thus Front CHOKe compares an incoming packet with the packet at the head of the buffer, while Back CHOKe compares it with the packet that arrived last. There is a slight further modification to Back CHOKe: it only drops incoming packets, but not packets from the buffer. This modification is not so much for analytical tractability (since dropping the packet from the back of the buffer is just as easy to analyze as Front CHOKe) as it is for studying aspects of the algorithm that Front CHOKe does not adequately model. The summaries are presented below. Details may be found in [16].

A. Front CHOKe

We model the system as a queue with N independent Poisson arrivals, each of rate λ_i. We assume that the service times are mutually independent, independent of the arrivals and exponentially distributed with mean $1/\mu$. The queueing discipline is first-in-first-out (FIFO).

The analysis invokes the PASTA[5]property and computes the goodput obtained by flow i to equal $\frac{\mu\lambda_i}{\mu+2\lambda_i}$. The condition for the queue to be stable is: $\sum_{i=1}^{N} \frac{\lambda_i}{\mu+2\lambda_i} < 1$, which due to packet dropping is much weaker than the usual condition: $\sum_{i=1}^{N} \lambda_i < 1$.

B. Back CHOKe

In Back CHOKe, the IDs of the M ($M < N$, where N is the total number of flows) recently admitted packets are stored in memory. The ID of every incoming packet is checked against these M IDs. If there is a match, then the incoming packet is dropped. This system is a Markov chain similar to a variant of the "Marching Soldiers Problem". The chain is shown to be reversible (for $M = 1$) or dynamically reversible (for $M > 1$) and its equilibrium distribution is computed. Various observations about its fairness properties are made[6].

V. CONCLUSIONS

This paper proposes a packet dropping scheme, CHOKe, which aims to approximate fair queueing at a minimal implementation overhead. Simulations suggest that it works well in protecting congestion-sensitive flows from congestion-insensitive or congestion-causing flows. Further work involves studying the performance of the algorithm under a wider range of parameters and network topologies, obtaining accurate theoretical models and insights, and considering hardware implementation issues.

REFERENCES

[1] Bertsekas, D. and Gallager, R., *Data Networks*, Second edition, Prentice Hall, 1992.

[2] Braden, B., Clark, D., Crowcroft, J., Davie, B., Deering, S.,Estrin, D., Floyd, S., Jacobson, V., Minshall, G., Partridge, C., Peterson, L.,Ramakrishnan,K., Shenker,S., Wroclawski, J., Zhang, L., "Recommendations on queue management and congestion avoidance in the internet", *IETF RFC (Informational) 2309*, April 1998.

[3] Demers, A., Keshav, S. and Shenker, S., "Analysis and simulation of a fair queueing algorithm", *Journal of Internetworking Research and Experience*, pp 3-26, Oct. 1990. Also in Proceedings of ACM SIGCOMM'89, pp 3-12.

[4] Floyd, S. and Jacobson, V., "Link-sharing and Resource Management Models for Packet Networks", *IEEE/ACM Transactions on Networking*, Vol. 3 No. 4, pp. 365-386, August 1995.

[5] Floyd, S. and Jacobson, V., "Random Early Detection Gateways for Congestion Avoidance", *IEEE/ACM Transaction on Networking*, 1(4), pp 397-413, Aug. 1993.

[6] Floyd, S., and Fall, K., "Router Mechanisms to Support End-to-End Congestion Control", *LBL Technical report*, February 1997.

[7] Floyd, S., and Fall, K., "Promoting the Use of End-to-End Congestion Control in the Internet", To appear in *IEEE/ACM Transactions on Networking*, August 1999.

[8] Floyd, S., Fall, K. and Tieu, K., "Estimating Arrival Rates from the RED Packet Drop History", *http://www.aciri.org/floyd/end2end-paper.html*, April 1998.

[9] Kelly, F., *Reversibility and Stochastic Networks*, John Wiley & Sons 1979.

[10] Keshav, S., "Congestion Control in Computer Networks", PhD Thesis, published as UC Berkeley TR-654 , September 1991.

[11] Lin, D. and Morris, R., "Dynamics of random early detection", *Proceedings of ACM SIGCOMM'97*, pp 127-137, Oct. 1997.

[12] Manin, A. and Ramakrishnan K., "Gateway Congestion Control Survey", *IETF RFC (Informational) 1254*, August 1991.

[13] McKenny, P., "Stochastic Fairness Queueing", *Proceedings of INFOCOM'90*, pp 733-740.

[14] Ott, T., Lakshman, T. and Wong, L., "SRED: Stabilized RED", *Proceedings of INFOCOM'99*, pp 1346-1355, March 1999.

[15] Pan, R. and Prabhakar, B., "CHOKe - A simple approach for providing Quality of Service through stateless approximation of fair queueing", *Stanford CSL Technical report CSL-TR-99-779*, March 1999.

[16] Pan, R., Prabhakar, B. and Psounis, K. "CHOKe - A stateless active queue management scheme for approximating fair bandwidth allocation" *Stanford CSL Technical report CSL-TR-99-787*, September 1999.

[17] Stoica, I., Shenker, S. and Zhang, H., "Core-Stateless Fair Queueing: Achieving Approximately Fair Bandwidth Allocations in High Speed Networks", *Proceedings of ACM SIGCOMM'98*.

[18] Suter, B., Lakshman, T., Stiliadis, D. and Choudhury, A., "Efficient Active Queue Management for Internet Routers", *Interop 98*.

[19] Wolff, R. *Stochastic Modeling and the Theory of Queues*, Prentice Hall 1989.

[20] ns - Network Simulator (Version 2.0), October 1998.

---

[5] PASTA: Poisson Arrivals See Time Averages

[6] Observe that the special case $N - M = 1$ results in a perfectly fair allocation through a round robin scheme.

Optimal routing to M parallel queues with no buffers

Eitan Altman
INRIA B.P. 93
2004 Route des Lucioles
06902 Sophia Antipolis Cedex
France
E-mail: altman@sophia.inria.fr

Sandjai Bhulai
Vrije Universiteit Amsterdam
De Boelelaan 1081a
1081 HV Amsterdam
The Netherlands
E-mail: sbhulai@cs.vu.nl

Bruno Gaujal
LORIA B.P. 102
54602 Villers-les-Nancy
France
E-mail: gaujal@loria.fr

Arie Hordijk
Leiden University
P.O. Box 9512
2300 RA Leiden
The Netherlands
E-mail: hordijk@wi.leidenuniv.nl

Abstract

In this paper we study the assignment of packets to M parallel heterogeneous servers with no buffer. The controller does not have knowledge on the state of the servers and bases all decisions on the number of time slots ago that packets have been sent to the different servers. The objective of the controller is to minimize the expected average cost.

We consider a general stationary arrival process, with no independence assumptions. We show that the problem can be transformed into a Markov Decision Process (MDP). We further show under quite general conditions on cost functions that only a finite number of states have to be considered in this MDP, which then implies the optimality of periodic policies.

For the case of two servers we obtain a more detailed structure of the cost and optimal policies. In particular we show that the average cost function is multimodular. Finally we present an application to optimal robot scheduling for web search engines.

Keywords: Markov Decision Processes, multimodularity, optimal open-loop control, routing.

1 Introduction

In a recent paper [8], Koole used the theory of MDPs with partial information to solve the problem of routing packets optimally into M servers with different service times. The objective was to minimize losses or maximize the throughput, and policies were restricted to those where no information is available on the state of the servers. For the special case of symmetrical servers and arrival processes, the optimality of the round-robin policy was established. For the case of 2 servers, it was shown that a periodic policy whose period has the form $(1, 2, 2, 2, \ldots, 2)$ is optimal, where 2 means sending a packet to the faster server. Similar results have been obtained in a dual model of server assignment in [6] with a somewhat different cost. The results in [6, 8] heavily depend on the Markovian structure of the arrivals.

An alternative approach based on the multimodularity and Schur convexity of the cost functions has been presented in [4]. It enabled us to use a novel theory for optimization of multimodular functions [7, 2, 3] in order to obtain structural results for the optimal policy without the Markovian assumptions made in [6, 8]. The objective was to minimize losses or maximize the throughput.

In this paper we extend our framework to general convex functions, not necessarily decreasing (as in [4]). We show that under quite general conditions, the problem can be transformed into an MDP with a *finite* state space, even when the arrival times have a general stationary distribution. No independence assumptions are required. We show that packets should be routed to every queue infinitely often, and that there exists an optimal periodic policy. We then present some further properties for the case of two queues. Finally, we apply this framework to the problem of robot scheduling for web search engines.

2 Problem formulation

Consider a system, to which packets arrive at times $(T_n)_{n\in\mathbb{N}}$. We use the convention that $T_1 = 0$ and we assume that the process $(\tau_n)_{n\in\mathbb{N}}$ of interarrival times $\tau_n = T_{n+1} - T_n$ is stationary[1].

Assume that the system has M parallel servers with no waiting room and independent service times, independent of the arrival process. Upon arrival of a packet a controller must route this packet to one of the M servers in the system. The service time has a general service distribution G_m when routed to server $m \in \{1, \ldots, M\}$. If there is still a packet present at the server, where the packet is routed to, then the packet in service is lost (we call this the preemptive discipline). In the special case of an exponential distribution, one can consider instead the non-preemptive discipline in which the arriving packet is lost; the results below will still hold.

We assume that the controller, who wishes to minimize some cost function, has no information on the state of the servers (busy or idle). The only information which the controller possesses is its own previous actions, i.e. to which server previous packets were routed. The mathematical formulation of this objective is given by the following T-horizon Markov decision process.

Let $\mathcal{X} = \left(\mathbb{N} \cup \{\infty\}\right)^M$ be the state space. The m^{th} component x_m of $x \in \mathcal{X}$ denotes the number of arrivals that occurred at other servers, since the last arrival to server m. Let $\mathcal{A} = \{1, \ldots, M\}$ be the action space, where action $a \in \mathcal{A}$ means that a packet is routed to server a. The transition probabilities are given by

---

[1] thus $(\tau_m, \ldots, \tau_{m+h})$ and $(\tau_n, \ldots, \tau_{n+h})$ are equal in distribution for all m, n and h.

$$p(x' \mid x, a) = \begin{cases} 1, & \text{if } x'_a = 1 \text{ and } x'_m = x_m + 1 \text{ for all } m \neq a \\ 0, & \text{otherwise.} \end{cases}$$

Define the direct costs by $c_t(x, a) = f_a(x_a)$, which reflects that the costs only depend on the chosen server and the state of that server for $t < T$. The terminal costs are given by $c_T(x) = \sum_a f_a(x_a)$ (note that the terminal costs use the same functions f_a). Defining these terminal costs will be essential for the mathematical results, as will be illustrated later in Example 3.4. It has also a natural physical interpretation, as will be illustrated in Remark 5.2.

The set of histories at epoch t of this Markov decision process is defined as the set $\mathcal{H}_t = (\mathcal{X} \times \mathcal{A})^{t-1} \times \mathcal{X}$. A policy π is a set of decision rules $(\pi_1, \pi_2, \dots)$ with $\pi_t : \mathcal{H}_t \to \mathcal{A}$. For each fixed policy π and each realization h_t of a history, the random variable A_t is given by $a_t = \pi_t(h_t)$. The random variable X_{t+1} takes values $x_{t+1} \in \mathcal{X}$ with probability $p(x_{t+1} \mid x_t, a_t)$. With these definitions the expected average cost criterion function $C(\pi)$ is defined by

$$C(\pi) = \limsup_{T\to\infty} \mathbb{E}_x^\pi \frac{1}{T} \sum_{t=1}^{T} c_t(X_t, A_t),$$

where $x = (x_1, \dots, x_M)$ is the initial state. Let Π denote the set of all policies. The Markov decision problem is to find a policy π^*, if it exists, such that $C(\pi^*) = \min\{C(\pi) \mid \pi \in \Pi\}$. This Markov decision model is characterized by the tuple $(\mathcal{X}, \mathcal{A}, p, c)$.

3 Structural results

In this section we investigate structural properties of the optimal policy. We assume that for all m, the cost functions f_m are convex. Moreover, they all satisfy either one of Conditions (1)–(3) below:

$$\lim_{x\to\infty} \big(f_m(x+1) - f_m(x)\big) = \infty. \tag{1}$$

$$f_m \text{ are strictly convex and } \lim_{x\to\infty} \big(f_m(x) - a_m x\big) = C. \tag{2}$$

(by strictly convex we mean that for all x, $f_m(x+2) - f_m(x+1) > f_m(x+1) - f(x)$).

$$f_m(x) = a_m x + C, \tag{3}$$

where $a_m \geq 0$ and C are constants (and C does not depend on m). Note that Condition (3) is not included in (2): it violates its first part. Condition (1) covers the case where f_m grows more than linearly, whereas (2)–(3) cover the case where f_m grows asymptotically linearly. These conditions are complementary to (1) since any one of them implies that $\lim_{x\to\infty} \big(f_m(x+1) - f_m(x)\big) < \infty$. In Conditions (1) and (2), f_m is strictly convex.

Theorem 3.1. *Assume that either one of Conditions (1)–(3) holds. Then*

(i) There exists an optimal policy that uses every server infinitely many times, thus $\sup\{j \mid \pi_j = m\} = \infty$ *for* $m \in \{1, \dots, M\}$.

(ii) There exists a periodic optimal policy.

Before proving the theorem we illustrate the necessity of the conditions (1)–(3). We present cost functions for which the above results do not hold.

Example 3.2. Consider the case of two queues with the costs

$$f_i(x) = a_i\, x + b_i \exp(-d_i\, x) + c_i, \quad i = 1, 2,$$

where $c_1 < c_2$, and where $b_i \geq 0$ and $d_i > 0$ are some constants. Then for a sufficiently small value of b_1, the policy that always routes to queue 1 is the only optimal one. We conclude that we cannot relax (2) or (3) and replace C by C_m.

Example 3.3. Consider the case of two queues with the costs $f_1(x) = a_1 x$ and $f_2(x) = \exp(-d_2 x)$. For $d_2 > 0$, the only optimal policy is the one that always routes to queue 1. This illustrates the necessity of the first part of condition (2). For $d_2 < 0$, the only optimal policy is the one that always routes to queue 2.

Next we present an example to illustrate the importance of the terminal cost.

Example 3.4. Assume that there are two queues, and the costs are given by $f_1(x) = x^2$ and $f_2(x) = 2\,x^2$, and assume that the terminal costs $c_T(x)$ were zero. Then the policy that always routes to queue 1 is optimal.

Proof of Theorem 3.1. First suppose that the cost function satisfies Condition (3). Interchanging assignments between any two queues for any finite horizon does not result in changes in cost for that horizon, due to the linearity of the cost function and the terminal cost. Hence any periodic policy that routes customers to all queues is optimal.

We consider next Conditions (1) and (2). Instead of describing a policy using a sequence π, we use an equivalent description using time distances between packets routed to each server. More precisely, given an initial state x, define the j^{th} instant at which a customer is routed to server m by

$$\eta^m(0) = -x_m \quad \text{and} \quad \eta^m(j) = \min\{\eta^m(j-1) < i \leq T \mid \pi_i = m\},$$

for $j \in \mathbb{N}$ (the min of an empty set is taken to be infinity.) Define the distance sequence δ^m by $\delta^m = (\delta^m_1, \delta^m_2, \ldots)$, with $\delta^m_j = \eta^m(j) - \eta^m(j-1)$, for $j \in \mathbb{N}$. (For simplicity we do not include the T in the notation.)

Let π be an arbitrary policy and m be an arbitrary server. Assume that the distance sequence $\delta = \delta^m$ for this server has the property that $\limsup_{T\to\infty}\{\delta_j \mid j \in \mathbb{N}\} = \infty$. We shall construct a policy π' with distance sequence $\delta^{\pi'}$ such that $\limsup_{T\to\infty}\{\delta^{\pi'}_j \mid j \in \mathbb{N}\}$ is finite and $C(\pi') \leq C(\pi)$.

Assume first that f satisfies Condition (1). Choose n_0 such that for all $n > n_0$

$$\min_{1\leq k\leq 2M+1} f_m(n+k) - f_m(k) - f_m(n)$$

$$> \max_{1\leq k<2M+1} f_l(2M+1) - f_l(k) - f_l(2M+1-k),$$

for all l. Since the supremum of the distance sequence is infinity, there is a j (and T) such that $\delta_j > n + 2M + 1$. Consider the $2M+1$ consecutive assignments starting at $\eta(j-1)$. Since there are M servers, it follows that there is at least one server, to which a packet is assigned three times during that period, say m'. Denote the distance (or interarrival) times of assignments to m' by δ^1 and δ^2. Replace the second assignment to m' in this sequence by an assignment to server m. Denote the new distance (or interarrival) times to m by δ_j' and δ_j'' (if $\eta(j) = T$ then the distance δ_j'' is not a real interarrival time). Consider the cost for a horizon of length l where l is an arbitrary integer larger than $\eta(j-1) + n_0 + 2M + 1$.

$$\begin{aligned}&\left[f_m(\delta_j) + f_{m'}(\delta^1) + f_{m'}(\delta^2)\right] - \left[f_m(\delta_j') + f_m(\delta_j'') + f_{m'}(\delta^1 + \delta^2)\right] = \\ &\left[f_m(\delta_j) - f_m(\delta_j') - f_m(\delta_j'')\right] - \left[f_{m'}(\delta^1 + \delta^2) - f_{m'}(\delta^1) - f_{m'}(\delta^2)\right] > 0,\end{aligned}$$

where the last inequality follows from the choice of n_0.

Consider now Condition (2). Since the supremum of the distance sequence is infinity, there is a j (and T) such that $\delta_j > 2n + 2M + 1$, for some n. Let $p = \min\{f_m(k) + a_m - f_m(k+1) \mid m = 1, \dots, M,\ k = 1, \dots, M\}$. Note that p is positive, since Condition (2) implies that $\big(f_m(x) - a_m x - C\big)$ is positive and strictly decreasing. Now choose n such that $2q = 2\big(f_m(n) - a_m n - C\big) < p$. Note that this is possible, since $f_m(n) - a_m n - C$ goes to zero as n goes to infinity. Consider the $2M + 1$ consecutive assignments starting n units after $\eta(j-1)$. There is at least one server, to which a packet is assigned three times, say m'. Replace the second assignment to m' in this sequence by an assignment to server m.

Define the function $g_i(k) = f_i(k) - a_i k - C$ for all i and consider the cost for a horizon of length l where l is an arbitrary integer larger than $\eta(j-1) + 2n + 2M + 1$. The decrease in cost due to the interchange is

$$\begin{aligned}&\left[f_m(\delta_j) + f_{m'}(\delta^1) + f_{m'}(\delta^2)\right] - \left[f_m(\delta_j') + f_m(\delta_j'') + f_{m'}(\delta^1 + \delta^2)\right] = \\ &\left[g_m(\delta_j) + g_{m'}(\delta^1) + g_{m'}(\delta^2)\right] - \left[g_m(\delta_j') + g_m(\delta_j'') + g_{m'}(\delta^1 + \delta^2)\right] > \\ &\left[g_m(\delta_j) + g_{m'}(\delta^1) + g_{m'}(\delta^2)\right] - \left[2g_m(n) + g_{m'}(\delta^1 + 1)\right] = \\ &\quad g_m(\delta_j) + g_{m'}(\delta^2) + \left[g_{m'}(\delta^1) - g_{m'}(\delta^1 + 1)\right] - \left[2g_m(n)\right] > \\ &\quad g_m(\delta_j) + g_{m'}(\delta^2) + p - 2q > 0,\end{aligned}$$

where δ^1, δ^2, δ_j' and δ_j'' are defined as before. The first inequality follows from the fact that $n < \delta_j'$, $n < \delta_j''$, $\delta^1 + 1 \leq \delta^1 + \delta^2$ and $f_m(x) - a_m x - C$ is decreasing. The second inequality follows from the definition of p. Since $f_m - a_m - C$ is positive it follows by construction of n that the last inequality holds.

Repeating the interchange procedure for every j for which $\delta_j > 2n+2M+1$ (when dealing with Condition (2)) or for which $\delta_j > n + 2M + 1$ (when dealing with Condition (1)) provides us a policy π' such that $C(\pi') \leq C(\pi)$ and $\sup\{\delta_j^{\pi'} \mid j \in \mathbb{N}\} < 2n + 2M + 1$. By repeating this procedure for every server, we get an optimal policy that visits a finite number of states. By Chapter 8 of Puterman [11] we know that the optimal policy can be chosen stationary. It follows that $\pi_n(h_n) = \pi_0(x_n)$. Since the state transitions are deterministic it follows that the optimal policy is periodic. □

Remark 3.5. One can show in fact a stronger result: there is a periodic policy which is optimal not only for the expected average cost but also for the finite horizon cost, for any length of the horizon. After a finite time, the state process is periodic when using that policy.

4 Multimodularity

Let $\mathbb{Z}$ be the set of integers. Let $e_i \in \mathbb{Z}^n$ for $i = 1, \ldots, n$ denote the vector having all entries zero except for a 1 in the i^{th} entry. Let d_i be given by $d_i = e_{i-1} - e_i$ for $i = 2, \ldots, n$. Then the base of vectors for multimodularity is defined as the collection $\mathcal{B} = \{b_0 = -e_1, b_1 = d_2, \ldots, b_{n-1} = d_n, b_n = e_n\}$. Following Hajek [7] a function g defined on $\mathbb{Z}^n$ is called multimodular if for all $x \in \mathbb{Z}^n$ and $b_i, b_j \in \mathcal{B}$ with $i \neq j$

$$g(x + b_i) + g(x + b_j) \geq g(x) + g(x + b_i + b_j). \tag{4}$$

We define the notion of an atom in order to study some properties of multimodularity. In $\mathbb{R}^n$ the convex hull of $n + 1$ affine independent points in $\mathbb{Z}^n$ forms a simplex. This simplex, defined on $x_0, \ldots, x_n$ is called an atom if for some permutation σ of $(0, \ldots, n)$

$$\begin{array}{rcrcl} x_1 & = & x_0 & + & b_{\sigma(0)}, \\ x_2 & = & x_1 & + & b_{\sigma(1)}, \\ \vdots & & \vdots & & \vdots \\ x_n & = & x_{n-1} & + & b_{\sigma(n-1)}, \\ x_0 & = & x_n & + & b_{\sigma(n)}. \end{array}$$

This atom is referred to as $A(x_0, \sigma)$ and the points $x_0, \ldots, x_n$ are called the extreme points of this atom. Each unit cube is partitioned into $n!$ atoms and all atoms together tile $\mathbb{R}^n$. We can use this to extend the function $g : \mathbb{Z}^n \to \mathbb{R}$ to the function $\bar{g} : \mathbb{R}^n \to \mathbb{R}$ as follows. If $x \in \mathbb{Z}^n$ then define the function by $\bar{g}(x) = g(x)$. Thus $\bar{g}$ agrees with g on $\mathbb{Z}^n$. If x is not an extreme point of an atom, then x is contained in some atom A. The value of $\bar{g}$ on such a point $x \in \mathbb{R}^n$ is obtained as the corresponding linear interpolation of the values of g on the extreme points of A. The following theorem shows the relation between multimodularity and convexity.

Theorem 4.1. *([7],[2], Theorem 2.1, [5]) A function g is multimodular if and only if the function $\bar{g}$ is convex.*

Let $g : \mathbb{Z}^n \to \mathbb{R}$ be an objective function in a mathematical model which has to be minimized. In general this is a hard problem to solve. However if the function g is multimodular, then we can use a local search algorithm, which converges to the globally optimal solution. In Koole and Van der Sluis [9] the local search algorithm has been proven for the search space $\mathbb{Z}^n$. The following theorem shows that it also holds for any convex subset S, which is a union of a set of atoms; thus in particular for $S = \mathbb{N}_0^n$ which is essential for our application. Multimodularity of g and convexity of $\bar{g}$ still hold when restricted to S. Multimodularity on this convex subset means that (4) must hold for all $x \in S$ and $b_i, b_j \in \mathcal{B}$ with $i \neq j$, such that $x + b_i$, $x + b_j$ and $x + b_i + b_j \in S$.

Theorem 4.2. *Let g be a multimodular function on S, a convex subset which is a union of atoms. A point $x \in S$ is a global minimum if $g(x) \leq g(y)$ for all $y \neq x$ which is an extreme point of $A(x, \sigma)$ for some σ.*

Proof. Let $x \in S$ be a fixed point. Now suppose that there is a $z \in \mathbb{R}^n$ such that $x + z \in S$ and $\bar{g}(x + z) < \bar{g}(x) = g(x)$. We show that there is an atom $A(x, \sigma)$ in S with an extreme point y such that $g(y) < g(x)$.

Since any n vectors of $\mathcal{B}$ form a basis of $\mathbb{R}^n$, we can write z as $z = \sum_{i=0}^{n} \beta_i b_i$. Furthermore, this can be done such that $\beta_i \geq 0$, since any $\beta_i b_i$ with $\beta_i < 0$ can be replaced by $-\beta_i \sum_{j=0,j\neq i}^{n} b_j$.

Now reorder the elements of $\mathcal{B}$ as $(b'_0, \ldots, b'_n)$ and the elements $(\beta_0, \ldots, \beta_n)$ as $(\beta'_0, \ldots, \beta'_n)$ such that $\beta'_0 \geq \cdots \geq \beta'_n \geq 0$ and $z = \sum_{i=0}^{n} \beta'_i b'_i$. Note that this notation is equivalent to the notation $z = \beta''_0 b'_0 + \beta''_1 (b'_0 + b'_1) + \cdots + \beta''_n (b'_0 + \cdots + b'_n)$ with all $\beta''_i \geq 0$ and with $b'_0 + \cdots + b'_n = 0$. Now fix a non-zero point $z' = \alpha z$ with $\alpha < 1$ such that $\alpha(\beta''_0 + \cdots + \beta''_{n-1}) \leq 1$. The set S is convex with x, $x + z \in S$, hence $x + z' \in S$. Since by Theorem 4.1 $\bar{g}$ is convex and $\bar{g}(x + z) < g(x)$, it follows that $\bar{g}(x + z') < g(x)$.

Let σ be the permutation induced by $(\beta''_0, \ldots, \beta''_n)$. Now consider the atom $A(x, \sigma)$, then by construction $x + z' \in A(x, \sigma)$. Since x is an extreme point and $\bar{g}$ is linear, there must be another extreme point, say y, such that $g(y) < g(x)$. ☐

In the previous section we showed that the optimal policy for all stationary arrival processes is periodic. In this section we shall show that the value function is multimodular when restricting to periodic policies. This can then be used to get a much more detailed structure of optimal policies (in particular, the regularity of optimal policies in the case of two queues), see [2]. Moreover the local search algorithm can be applied to find an optimal policy.

We note that some results on multimodularity have been obtained already in [4] for each individual queue with respect to the routing sequence to that queue. This allows to obtain some structure for the optimal policies as in [3]. The novelty of the results here is that, for the case of two queues to which we restrict, multimodularity is obtained directly for the global cost of all queues rather than for each one separately.

The notation of the distance sequence can be beneficially used to approach the decision problem. After the first assignment to server m, the distance sequence δ^m for server m is periodic, say with period $d(m)$. Therefore in future discussions we will write $\pi = (\pi_1, \ldots, \pi_n)$ for the periodic assignment sequence with period n and with a slight abuse of notation we denote the periodic distance sequence for server m by $\delta^m = (\delta^m_1, \ldots, \delta^m_{d(m)})$.

The periodicity reduces the cost function in complexity. Since we use the expected average cost function, we only have to consider the costs incurred during one period. It is tempting to formulate that the cost function is multimodular for all m. Unfortunately it is not clear that this is true. However when we assume that $M = 2$ then we are able to prove multimodularity. The cost function for server m is given by $g'_m(\pi)$

$$g'_m(\pi) = \frac{1}{n} \sum_{j=1}^{d(1)} f_m(\delta^1_j). \tag{5}$$

This function only looks at the distance sequence assigned to the first queue. Note that $g(\pi)$ can now be expressed as $g(\pi) = g'_1(\pi) + g'_2(\vec{3} - \pi)$. ($\vec{3}$ is the vector whose components are all 3.) We first prove that $g'_m(\pi)$ is multimodular in π. Then multimodularity of $g(\pi)$ follows as the sum of two multimodular functions.

Lemma 4.3. *Assume that f_m are convex. Let π be a fixed periodic policy with period n. Let $g'_m(\pi)$ be defined as in (5). Then $g'_m(\pi)$ is a multimodular function in π.*

Proof. Since π is a periodic sequence, the distance sequence $\delta = \delta^1$ is also a periodic function, say with period $p = d(1)$. Now, define the function h_j for $j = 1, \ldots, p$ by $h_j(\pi) = f_m(\delta_j)$. The function h_j represents the cost of the $(j + 1)^{\text{st}}$ assignment to server m by looking at the j^{th} interarrival time. We will first check the conditions for multimodularity for $\mathcal{V} = \{b_1, \dots, b_{n-1}\}$.

Let v, $w \in \mathcal{V}$ with $v \neq w$. If none of these elements changes the length of the j^{th} interarrival time then $h_j(\pi) = h_j(\pi + v) = h_j(\pi + w) = h_j(\pi + v + w)$. Suppose that only one of the elements changes the length of the interarrival time, say v, then $h_j(\pi + v) = h_j(\pi + v + w)$ and $h_j(\pi) = h_j(\pi + w)$. In both cases the function $h_j(\pi)$ satisfies the conditions for multimodularity.

Now suppose that v adds and w decreases the length of the j^{th} interarrival time by one. Then $\delta_j^1(\pi + v) - \delta_j^1(\pi) = \delta_j^1(\pi + v + w) - \delta_j^1(\pi + w)$. Since h_j is a convex function, it follows that $h_j(\pi + w) - h_j(\pi + v + w) \geq h_j(\pi) - h_j(\pi + v)$. Now the multimodularity condition in Equation (4) directly follows by rearranging the terms. Since $g'_m(\pi)$ is a sum of $h_j(\pi)$ it follows that $g'_m(\pi)$ is multimodular for $\mathcal{V}$.

Now consider the elements b_0 and b_n and note that the application of b_0 and b_n to π splits an interarrival period and merges two interarrival periods respectively. Therefore

$$\begin{aligned}
n\, g'_m(\pi + b_0) &= n\, g'_m(\pi) - f_m(\delta_1) - f_m(\delta_p) + f_m(\delta_1 + \delta_p), \\
n\, g'_m(\pi + b_n) &= n\, g'_m(\pi) - f_m(\delta_p) + f_m(\delta_p - 1) + f_m(1), \\
n\, g'_m(\pi + b_0 + b_n) &= n\, g'_m(\pi) - f_m(\delta_1) - f_m(\delta_p) + f_m(\delta_1 + 1) + f_m(\delta_p - 1).
\end{aligned}$$

Now $n\big[g'_m(\pi+b_0)+g'_m(\pi+b_n)-g'_m(\pi)-g'_m(\pi+b_0+b_n)\big] = \big[f_m(\delta_1+\delta_p)+f_m(1)\big] - \big[f_m(\delta_1 + 1) + f_m(\delta_p)\big]$. Let $k = \delta_1 + \delta_p + 1$. Since the function $f_m(x) + f_m(y)$ with $x + y = k$ is a symmetric and convex function, it follows from Proposition C2 of Chapter 3 of Marshall and Olkin [10], that $f_m(x) + f_m(y)$ is also Schur-convex. Since $(\delta_1 + 1, \delta_p) \prec (\delta_1 + \delta_p, 1)$, the quantity above is non-negative.

In the case that we use $w = b_0$ and $v \in \mathcal{V}$ such that v does not alter δ_1, then it follows that $g'_m(\pi + v + w) = g'_m(\pi + v) + g'_m(\pi + w) - g'_m(\pi)$. The same holds for $w = b_n$ and $v \in \mathcal{V}$ such that v does not alter δ_p. Suppose that v does alter δ_1, then we have $n\big[g'_m(\pi + b_0) + g'_m(\pi + v) - g'_m(\pi) - g'_m(\pi + b_0 + v)\big] = \big[f_m(\delta_1 + \delta_p) + f_m(\delta_1 - 1)\big] - \big[f_m(\delta_1 + \delta_p - 1) + f_m(\delta_1)\big]$. When v alters δ_p we have $n\big[g'_m(\pi + b_n) + g'_m(\pi + v) - g'_m(\pi) - g'_m(\pi + b_n + v)\big] = \big[f_m(\delta_p + 1) + f_m(l)\big] - \big[f_m(\delta_p) + f_m(l + 1)\big]$ for some $l < \delta_p$. Now by applying the same argument as in the case of b_0 and b_n we derive multimodularity of $g'_m(\pi)$ for the base $\mathcal{B}$. ☐

5 Robot scheduling for web search engines

In [4] we specified a routing problem where the expected average weighted loss rate was to be minimized (or equivalently, the average weighted throughput or average weighted number of packets at service was to be maximized). This gave rise to an immediate cost of the form:

$$c(x, a) = \mathbb{E} \exp \left[-\mu_i \sum_{i=1}^{x_a} \tau_i \right].$$

Due to stationarity of the interarrival times, this cost function satisfies Condition (2), so the results of the previous sections apply. In this section we present another application studied in Coffman, Liu and Weber [6] under assumptions that are more restrictive than ours.

The World Wide Web offers search engines, such as Altavista, Lycos, Infoseek and Yahoo, that serve as a database that allow to search information on the web. These search engines often use robots that periodically traverse a part of the web structure in order to keep the database up-to-date.

We consider a problem where we assume that there is a fixed number of M web-pages. The contents of page i is modified at time epochs that follow a Poisson process with parameter μ_i. The time a page is considered up-to-date by the search engine is the time since the last visit by the robot until the next time instant of modification; at this point the web-page is considered out-of-date until the next time it is visited by the robot. The times between updates by the robot are given by a sequence τ_n. In Coffman, Liu and Weber [6] these times are assumed to be i.i.d., but in our framework we may allow them to form a general stationary sequence.

Let r_i denote the obsolescence rate of page i, i.e. the fraction of time that page i is out-of-date. Then the problem is to find an optimal visiting schedule such that the sum of the obsolescence rates r_i weighted by specified constants c_i is minimized. A reasonable choice for the weights c_i would be the customer page-access frequency, because the total cost then represents the customer total error rate. The case where the customer access frequency $c_i = k\,\mu_i$ is proportional to the page-change rate μ_i is reasonable under this interpretation, since the greater the interest for a particular page is, the more likely the frequency of page modification is.

We now show that this problem is equivalent to the problem studied in Section 2. Indeed, the robot can be considered as the controller in the previous problem. An assignment of the n^{th} packet to server i corresponds with sending the robot to page i and requires τ_n time in order to complete the update. The duration of consecutive modifications of page i correspond to the service times of a packet. When a packet before server i is being served then page i is considered to be up-to-date and after service the page is considered out-of-date.

Let S_i be an exponential random variable with parameter μ_i. The cost which one incurs when sending a packet to server a should reflect the expected obsolescence time. Some straightforward computations yield

$$c(x,a) = k\,\mu_a \mathbb{E}\left[\sum_{i=1}^{x_a} \tau_i - S_a\right]^+ = k\,\mu_a\,\mathbb{E}\left[\sum_{i=1}^{x_a} \tau_i - S_a\right] + k\,\mu_a\,\mathbb{E}\left[S_a - \sum_{i=1}^{x_a} \tau_i\right]^+$$

$$= k\,\mu_a\,x_a \mathbb{E}\,\tau_1 + k\left[\mathbb{E}\exp\left(-\mu_a \sum_{i=1}^{x_a} \tau_i\right) - 1\right].$$

This cost function clearly satisfies Condition (2). Hence the theorems from the previous sections can indeed be applied to this scheduling problem.

Remark 5.1. The assumption that the weights c_i are proportional to the page-change rates μ_i is essential in this problem. The cost function for general c_i is given by

$$c(x,a) = c_a\,x_a \mathbb{E}\,\tau_1 + \frac{c_a}{\mu_a}\left[\mathbb{E}\exp\left(-\mu_a \sum_{i=1}^{x_a} \tau_i\right) - 1\right].$$

When the c_i are not proportional to the page-change rates, then the cost function is of the type mentioned in Example 3.2. Therefore if for some i, $\frac{c_i}{\mu_i}$ is sufficiently large (in comparison to others) then it becomes optimal never to update page i. This is an undesirable situation and it shows that the problem is not always well posed when the costs are not proportional to the page-change rates μ_i.

Remark 5.2. This problem of robot scheduling for web search engines illustrates the importance of including the terminal cost c_T in terms of modeling. Indeed, for any finite horizon T, if we wish that the cost represents indeed the obsolescence time of a page, we have to make sure that if this page is never updated (or at least it stops to be updated after some time), this will be reflected in the cost. It is easy to see that the terminal cost defined in Section 2 indeed takes care of that.

References

[1] E. Altman, S. Bhulai, B. Gaujal and A. Hordijk. Optimal routing problems and multimodularity. Research Report RR–3727, INRIA, 1999.

[2] E. Altman, B. Gaujal and A. Hordijk. Multimodularity, convexity and optimization properties. Technical Report TW–97–07, Leiden University, 1997.

[3] E. Altman, B. Gaujal and A. Hordijk. Balanced sequences and optimal routing. Research Report RR–3180, INRIA, 1997.

[4] E. Altman, B. Gaujal, A. Hordijk and G Koole. Optimal admission, routing and service assignment control: the case of single buffer queues. *IEEE Conference on Decision and Control*, Dec. '98, Tampa, Florida, USA.

[5] M. Bartroli and S. Stidham, Jr. Towards a unified theory of structure of optimal policies for control of network of queues. Technical Report, Department of Operations Research, University of North Carolina, Chapel Hill, 1987.

[6] E.G. Coffman Jr., Z. Liu and R.R. Weber. Optimal robot scheduling for web search engines. *Journal of Scheduling*, Vol. 1, 1999.

[7] B. Hajek. Extremal splitting of point processes. *Mathematics of Operations Research*, 10:543–556, 1985.

[8] G. Koole. On the static assignment to parallel servers. *IEEE Transactions on Automatic Control*, 1999. (to be published).

[9] G. Koole and E. van der Sluis. An optimal local search procedure for manpower scheduling in call centers. Vrije Universiteit Amsterdam.

[10] A.W. Marshall and I. Olkin. *Inequalities: Theory of Majorization and its Applications.* Academic Press, 1979.

[11] M.L. Puterman. *Markov Decision Processes: Discrete Stochastic Dynamic Programming.* John Wiley & Sons, 1994.

The Effect of Scale on Internet Quality

Matthew Siler and Jean Walrand

Department of Electrical Engineering and Computer Science
University of California at Berkeley
Berkeley, CA 94720
{siler,wlr}@eecs.berkeley.edu

Abstract

In this paper, we investigate the performance of TCP when a random number of connections share the bottleneck link. The statistical fluctuation of active TCP connections can have a profound effect on Internet quality, and requires an overprovisioning of link capacities. We show, through a simple link-sharing model, that access links must be significantly overprovisioned compared to backbone links to assure quality. Furthermore, access control algorithms become considerably less complex within the backbone because the number of active connections, although random, exhibits less statistical fluctuation. In particular, we look at how admission control and weight provisioning algorithms in DiffServ improve with scale.

1 Introduction

Today, one of the greatest challenges facing network engineers is to design networks that meet or exceed user expectations of performance. As these expectations increase, a central question is how network infrastructure and control mechanisms should scale with demand. Within the current Internet, the quality of service provided to TCP connections ultimately depends on a large number of factors, including network topology, routing, propagation delays, and the number of active connections. For example, the network topology and routing decisions determine the number of nodes and the link capacities that each TCP connection traverses. Furthermore, propagation delays and the number of active connections along each route decide, to a certain extent, the achievable throughput per TCP connection. Often, a detailed understanding of performance requires that the dynamics of TCP be modeled at the packet level [8]. However, the random fluctuation of the number of TCP connections at the bottleneck link can also have a profound effect on TCP performance and is not adequately captured by these models.

In this paper, we model the TCP transaction level where it is easier to quantify how TCP performance relates to the number of active connections. We use this model to investigate how network provisioning and control algorithms scale with the number of users to assure quality with high probability. In particular, we show that access links must be significantly overprovisioned, compared to backbone links, to handle statistical fluctuations in the number of active connections. Furthermore, for real-time traffic using UDP, access control may be eased within the backbone provided that tight control is maintained at the access links.

The paper is organized as follows. In Section 2, we investigate the impact of a random number of TCP connections on performance in a single link. This model shows the dramatic improvement of quality with scale, and demonstrates the degree to which access links must be overprovisioned to provide quality to users. Furthermore, we investigate the effect of the bias of TCP towards connections with short round-trip times, and the corresponding impact on performance. Second, we apply this model to an arbitrary network to investigate how the network scales to meet end-to-end performance in Section 3. In particular, we outline an incremental approach to network provisioning that increases the capacity of the most likely first bottleneck link. Third, in Section 4, we extend this model to the proposed Differentiated Services (DiffServ) [4] architecture for providing per-class performance guarantees. In particular, we consider a high-priority Premium Service (PS), for voice-over-IP traffic, and an Assured Service (AS), for multiclass TCP traffic, which provides a relative rate to each AS class using Weighted Fair Queuing (WFQ) [1]. Here, we investigate how admission control, for VoIP traffic, and weight provisioning, for multiclass TCP traffic, scale in the DiffServ network.

2 TCP Performance: A Transaction-level Approach

The TCP protocol suite uses a collection of algorithms to back-off during periods of network congestion and to speed-up when bandwidth becomes available [3]. The complex feedback mechanism of TCP introduces correlations between traffic streams that are difficult to model. Instead, we model TCP at the transaction level, rather than the packet level, which takes into account the statistical fluctuation of the number of active TCP connections. In this section, we consider a transaction-level TCP and link-sharing model that focuses on how performance scales with the number of users.

2.1 Link-Sharing Model

A great deal of literature has been devoted to the study of TCP performance. Typically, these papers focus on the dynamics of the congestion recovery mechanisms of TCP [8]. However, such analysis often relies on the number of connections being fixed when in fact this is random process. The statistical fluctuation of the number of TCP connections can have a significant impact on quality in the absence of admission control.

We propose a simple model of the TCP connection process. Suppose we have a large number of TCP users, where a fraction of them are active at any time. Furthermore, users act independently of one another, and each user is active with a relatively small probability. Then, we can model the number of active transactions reasonably well with a Poisson random variable.

Consider a single link, with capacity C, that is shared by N TCP connections. If the link achieves a "fair sharing" of the link, then each TCP connection transmits at rate R, where

$$R = \frac{C}{N}. \tag{1}$$

This model presents an optimistic view of TCP, which we call *idealized TCP*. That is, we make the following assumptions:

- Every TCP connection receives its "fair share" of bandwidth.
- All TCP connections immediately settle into equilibrium.
- TCP efficiently uses the full bandwidth of the link.

Of course, these are ideal conditions, but they give us a useful benchmark of optimal performance. There are two departures from this model that must be considered.

First, the most widely deployed version of TCP, TCP-Reno, aggressively grabs unused bandwidth and is not very network-friendly [6]. In the congestion avoidance phase, TCP-Reno increases its window size by one during each round-trip time, so connections with a shorter round-trip time will be able to take more than their fair share of bandwidth.

Second, the dynamics of TCP prevent it from taking full advantage of the available rate. One reason is the transient state of TCP requires that connections resettle to new rates whenever a connection either starts or finishes. Since we look at providing minimum rates, this effect is not expected to be significant in provisioning, but may be important if other performance measures are used. Furthermore, the efficiency of TCP depends largely on maintaining a constant backlog in the link queue. However, TCP is not able to achieve this except with very large buffers [7] that absorb the fluctuations of the source rate, but also increase the queuing delay.

Our model of TCP transactions allows us to understand how performance relates to network scale, but requires a number of refinements to be more precise. We study the impact of one of these modeling uncertainties – the effect of TCP bias – from the idealized TCP model in Section 2.3. The effect of the transient state of TCP and of the efficiency of TCP are subjects of further study.

2.2 Link Capacity Provisioning

To meet the performance expectations of users, the network must be able to control the network. For short-lived connections, such as Web traffic, one approach is to provision networks so that poor performance is unlikely. In our link-sharing model, the performance depends upon the statistical fluctuations of the number of active connections. An important question is how C should be provisioned to scale with the number of users.

Suppose that we want to provision the link with capacity C so that we guarantee a rate of r to each TCP connection with high probability. That is,

$$P(R \geq r) \geq 1 - \gamma \qquad (2)$$

where $\gamma = 10^{-3}$. We model the number of active transactions, N, as a Poisson random variable with rate parameter $\lambda = n\alpha$, where n is the number of independent users and α is the probability that a user is active. For large n, we may approximate N with a Gaussian random variable,

$$N \approx \lambda \left(1 + \frac{Z}{\sqrt{\lambda}}\right),$$

where Z is a normalized Gaussian random variable.

Using this Gaussian approximation, a statistical guarantee of γ requires that we provision C to be

$$C \geq r\lambda \left(1 + \frac{Q^{-1}(\gamma)}{\sqrt{\lambda}}\right)$$

where $Q(x) \equiv P(\mathcal{N}(0,1) > x)$. Then, the link must be overprovisioned by a factor of $\frac{Q^{-1}(\gamma)}{\sqrt{\lambda}}$ to accommodate the statistical fluctuation of active users. The degree of overprovisioning of the link decreases on the order of $\frac{1}{\sqrt{n}}$ as n increases. In the backbone, where n is expected to be large, the link can be highly utilized and still provide a level of performance that meets user expectations. However, on an access link, where n is expected to be much smaller, and the statistical fluctuation of users is much larger, the link must be significantly overprovisioned to achieve this same performance.

As an example, consider an access link with $n = 500$ and $\alpha = 2 \times 10^{-2}$, so that, on average, 10 users are active at any time. For a guarantee of $r = 100$ kbps, the capacity must be $C \approx 2$ Mbps and is overprovisioned by a factor of 2. However, if $n = 50,000$, as can be expected within a backbone link, then $C \approx 110$ Mbps and the link is only overprovisioned by 10%. The implication of this result is that acceptable quality is achievable with statistical multiplexing, but requires proper overprovisioning of the link capacities.

2.3 TCP Bias

In the previous section, we studied a network with an idealized version of TCP. In practice, the current implementation of TCP and IP routers does not allow the network to achieve this. A significant problem is the bias of TCP towards connections with short round-trip times.

Suppose that n total users share a link, and the active users are divided into two groups. A fraction α_i of the users are active and have a round-trip time of rtt_i. The rate of a TCP connection that shares a bottleneck link with other connections is roughly proportional to the inverse of the round-trip time. Therefore, if R_i is the rate of users in group i, then $R_1 \approx \beta R_2$ where $\beta = \frac{rtt_2}{rtt_1}$. Furthermore, if N_i is the number of active users of group i, then $R_2 \approx \frac{C}{\beta N_1 + N_2}$. That is, to group 2 users, there appear to be $(\beta - 1)$ times more users of group 1 that equally share the link than actually exist. In this case, to provide the same statistical guarantee of rate r to all users, we require C to be

$$C \geq r(\beta\lambda_1 + \lambda_2)\left(1 + \frac{Q^{-1}(\gamma)}{\sqrt{\beta\lambda_1 + \lambda_2}}\right)$$

where $\lambda_i = n\alpha_i$. From this model, we see that the effect of TCP bias on quality may be significant. For example, for $\beta = 2$, $\lambda_1 = \lambda_2 = 5$, and $r = 100$ kbps, a capacity of 2.7 Mbps is needed, whereas only 2 Mbps is needed when no bias is present. In this case, 26% of the capacity is needed to account for the bias of TCP. For large n, the fraction of the capacity that must be provisioned due to bias is approximately $\frac{\beta-1}{\beta+\lambda_2/\lambda_1}$, which becomes significant for large β.

One solution to the problem is to use a TDM scheduler to separate the two groups of users. In this case, the capacity C should be

$$C \geq r(\lambda_1 + \lambda_2)\left(1 + \frac{(\sqrt{\lambda_1} + \sqrt{\lambda_2})Q^{-1}(\gamma)}{\lambda_1 + \lambda_2}\right).$$

This removes the effect of TCP bias, but also reduces the statistical multiplexing (since TDM does not allow link sharing between groups of users). Other solutions involve making modifications to TCP, buffer management, or scheduling algorithms used in the Internet. Newer versions of TCP, such as TCP-Vegas, mitigate the effect of TCP bias by estimating available bandwidth through measurements of queuing delays [6]. Furthermore, RED gateways, ECN packet marking, and WFQ scheduling have been proposed as mechanisms for reducing this bias. Using this model, it is possible to evaluate the benefit of implementing these mechanisms.

3 TCP Performance across the Network

In the previous section, we introduced an idealized model of TCP performance that achieves a "fair sharing" of a single link. In this section, we extend this model to a network that achieves max-min fairness. Essentially, max-min fair sharing dictates that each connection gets at least an equal share of its bottleneck link.

3.1 Network Model

We model the network as an arbitrary topology of K routers with J unidirectional links connecting them. For any path p in the network, let $A(p, j) = 1$ if link j is on path p, and zero otherwise. The transmission rate of link j is $C(j)$. Now, suppose that on path p there are $N(p)$ TCP connections that are active, where $N(p)$ is Poisson with rate $\lambda(p)$. Note that this model is robust enough to include multiple classes of traffic, since more than one path may traverse the same set of physical links. An important variable in our model is $M(j)$, the number of TCP connections that share a physical link j:

$$M(j) = \sum_p N(p)A(p, j).$$

We assume that the routes are static, so that the packets of each TCP connection follow the same path. For routing metrics based on number of hops or link capacities, this is a realistic model since these values change only when the network topology does. However, if the routing metric depends upon the load of each link, then the chosen routes of each TCP connection will become correlated.

3.2 Max-Min Fairness

Since the TCP protocol does not differentiate between types of applications, the goal of TCP traditionally has been to achieve a fair sharing of network resources amongst active connections. In particular, max-min fairness requires that each connection be allotted at least its equal share of its bottleneck link. A link j is said to be a bottleneck link if, by increasing $C(j)$, we can increase the throughput of at least one connection that traverses j (see [5] for details).

The rate per connection along path p, given by $R(p)$, can be computed recursively starting with the first bottleneck in the network. Let b be a vector of link indices, with b_1 the first bottleneck in the network. The first bottleneck is computed as

$$b_1 = \arg\min_j \frac{C(j)}{M(j)}.$$

That is, it is the link where the fair share of bandwidth per connection is smallest. However, this minimum may not be unique – there may be more than one link in the network that exhibits this minimum fair share of bandwidth. In this case, any of these links may be chosen as the first bottleneck since it will not affect the achievable rate per connection. For connections that traverse this bottleneck link, such that $b_1 \in p$, the corresponding rate per connection is

$$R(p) = \frac{C(b_1)}{M(b_1)}.$$

The other bottlenecks may be computed recursively. Let $B^{(i)}$ be the set of the first i bottleneck links, where $B^{(i)} = \{b_1, \ldots, b_i\}$. Then, the i-th bottleneck, for $i \geq 2$ may be computed from the following equation:

$$b_i = \arg \min_{j \notin B^{(i-1)}} \frac{C(j) - \sum_p R(p)N(p)A(p,j)\left(1 - \prod_{k \in B^{(i-1)}}(1 - A(p,k))\right)}{M(j) - \sum_p N(p)A(p,j)\left(1 - \prod_{k \in B^{(i-1)}}(1 - A(p,k))\right)}.$$

Since every connection goes through at least one bottleneck link, we can define $b^*(p)$ to be the bottleneck link for all connections on path p, i.e. $b^*(p) = \min\{i : b_i \in p\}$. This leads to a rate per connection on path p of

$$R(p) = \frac{C(b^*(p))}{M(b^*(p))}.$$

3.3 Network Capacity Provisioning

The objective of provisioning capacities in a network is to provide end-to-end performance guarantees to TCP connections. A straightforward approach is to choose $C(\cdot)$ such that

$$P\left(\frac{C(j)}{M(j)} \geq r\right) = 1 - \gamma$$

for all $j = 1, \ldots, J$. If we let $\psi(j) \equiv \sum_p \lambda(p)A(p,j)$, then

$$C(j) := r\psi(j)\left(1 + \frac{Q^{-1}(\gamma)}{\sqrt{\psi(j)}}\right), \tag{3}$$

and the probability that a rate of at least r is available on path p is given by

$$P\left(R(p) \geq r\right) = P\left(\sum_{p'} N(p')A(p',j) \leq \frac{C(j)}{r}, \forall j \in p\right). \tag{4}$$

Since, for all $j \in p$, these events are positively correlated under our modeling assumptions, we can lower bound this probability. Combining this with the Gaussian approximation of $N(p)$, we have the following bound:

$$P\left(R(p) \geq r\right) \geq \prod_{j \in p}\left(1 - Q\left(\frac{C(j)/r - \psi(j)}{\sqrt{\psi(j)}}\right)\right).$$

Using the rate assignment given by (3) above,

$$P\left(R(p) \geq r\right) \geq \prod_{j \in p}(1 - \gamma),$$

which is tight for small γ. It is also possible to refine this bound by computing (4) using numerical integration methods.

If link capacities are provisioned incrementally, then one possible heuristic is to increase the capacity of the most likely first bottleneck link. For a fixed number of TCP connections along each path, the max-min equilibrium state is easy to identify. However, since the number of connections is random, so are the locations of the bottlenecks and the minimum rates per connection. From our model, the probability that link j is the first bottleneck is

$$P(b_1 = j) = P\left(\frac{C(j)}{M(j)} < \frac{C(i)}{M(i)}, \forall i \neq j\right).$$

For $J = 2$, this probability can be easily evaluated, but requires numerical integration for larger networks. Furthermore, if changing the capacity of the links does not affect the routing of TCP connections, then it is possible to evaluate the effect on end-to-end performance along any path p by computing (4) with the new link capacity.

4 Differentiated Services

Although the Internet has primarily been used for data networking, the desire for real-time applications has focused attention on providing differentiated services within the Internet. Furthermore, there is also interest in providing different classes of service to data connections, such as Web sessions, to improve the quality of selected applications. DiffServ places a limited amount of control into edge and backbone routers to support some differentiation of services for aggregate connections. Limiting control to DiffServ classes, rather than individual flows, provides a scalable approach to providing quality on the Internet. In this section, we consider a DiffServ network that provides services to voice-over-IP and multiclass TCP traffic. In particular, we look at the importance of admission control and provisioning in both the access and backbone nodes.

4.1 DiffServ Model

4.1.1 Premium Service Class

The first large-scale deployment of a real-time application is expected to be voice-over-IP. Most VoIP applications use UDP rather than TCP as a transport protocol over IP. Since TCP is a closed-loop protocol, the rate of TCP connections depends upon the state of the network. However, VoIP calls typically use open-loop transmission protocols built on UDP that send at rates independent of the state of the network. Furthermore, these applications tend to be more long-lived and can withstand some loss but can not tolerate large delays, so lost packets are not retransmitted. The proposed architecture for VoIP traffic is that it use a Premium Service, where traffic is policed and forwarded with high priority at each DiffServ node.

We denote the number of VoIP connections active on path p as $N_v(p)$, which is a Poisson random variable with rate parameter $\lambda_v(p)$. We model each active voice connection as an on-off Markov process, where the transmission rate is h when the voice connection is "on", and zero otherwise. Furthermore, we let $M_v(j)$ be the number of active voice connections on link j, which is Poisson with rate $\sum_p \lambda_v(p)A(p,j)$. Suppose that the probability that an active voice connection is "on" is $0 < q < 1$. Then, at any given time, the number of voice connections that are both active and "on" is $N_v^{on}(p)$, which has a Poisson distribution with rate $q\lambda_v(p)$.

Since VoIP packets are given high priority in DiffServ nodes, we employ a bufferless model for the Premium Service and focus on the "instantaneous" transmission rate of voice connections over path p. If we ignore the propagation delays between nodes, and the reshaping of PS traffic at each node, then the rate of PS traffic at each node along path p is $R_v(p) = h \cdot N_v^{on}(p)$.

4.1.2 Assured Service Class

Data applications are expected to take advantage of the unused bandwidth of the PS class using the AS class. Traffic that is in the AS class is treated with low priority with respect to the PS class, but receives better service than best-effort traffic. Since admission control is not used for AS traffic, we propose a simple service commitment model that provides a rate $R_d(p_k)$ to each user in class k along path p_k that scales with the number of users. For example, with two classes, we may provision the network such that $R_d(p_1) \approx 2R_d(p_2)$, so that the users in class 1 see a rate that is twice as large as users in class 2 provided that p_1 and p_2 traverse the same route.

4.2 VoIP Provisioning and Admission Control

The introduction of differentiated services on the Internet will require more sophisticated management and control mechanisms. For VoIP traffic, the clear superiority of the PS class over the AS class requires that we limit the number of admitted connections, requiring some form of admission control. Furthermore, the acceptance of Internet telephony will depend largely on whether the packet-switched network can provide equivalent service to the circuit-switched network at a reduced cost. Therefore, provisioning the Internet for voice calls requires that we design the capacity of the network sufficiently large to support the user population by keeping the blocking probability small, but also take advantage of the statistical multiplexing of voice and data connections.

In a circuit-switched network, the blocking probability may be estimated using a fixed-point Erlang approximation. With Internet telephony, dedicated circuits are replaced with virtual circuits using the PS class. Therefore, we may use a similar approach to compute $\bar{m}_{v,j}$, the number of voice calls that must be admitted at link j to support a target blocking probability of η_v, from the network topology and statistics of voice call usage.

For circuit-switched networks, we provision the link capacity to be $C(j) = h \cdot \bar{m}_{v,j}$ to achieve the target blocking probability for new calls. However, Since VoIP applications can withstand some loss, we may provision $C(j)$ to be substantially smaller depending upon the statistical multiplexing of voice connections. For m admitted voice connections, the number of connections that are "on" is binomially distributed. For large $\bar{m}_{v,j}$, we approximate this with a Gaussian random variable with mean $\bar{m}_{v,j}q$ and variance $\bar{m}_{v,j}q(1-q)$. Therefore, we choose $C(j)$ such that

$$C(j) \geq \bar{m}_{v,j}hq\left(1 + \frac{Q^{-1}(\gamma_v)}{\sqrt{\frac{\bar{m}_{v,j}q}{1-q}}}\right), \tag{5}$$

where γ_v is the loss probability for voice traffic. This ensures that the blocking probability and loss probability of admitted users is within acceptable bounds. It is possible to refine the estimate of $C(j)$ using other asymptotic approaches, such as large deviations, especially when γ_v is smaller than 10^{-3}. If the link capacities $C(\cdot)$ are fixed, then the problem is to compute $\bar{m}_{v,j}$ such that the performance objectives for admitted voice calls are met. That is, at each link j, we compute

$$\bar{m}_{v,j} = \max\left\{m : Q\left(\frac{C(j)/h - mq}{\sqrt{mq(1-q)}}\right) \leq \gamma_v\right\}. \tag{6}$$

In either case, the admission control algorithm requires that each new voice call request a circuit from each link in the chosen path. That is, a new call on path p may be added if $M_v(j) + 1 \leq \bar{m}_{v,j}$ for every $j \in p$ [2].

However, it may not be necessary to perform admission control at each link if the network is properly provisioned. For example, consider a network with J access links, with line rates $c_1, c_2, \ldots, c_J$, that are being multiplexed onto a single backbone link with capacity c. Using (5), we can provision c_j such that $\bar{m}_{v,j}$ VoIP calls may be admitted to access link j with a blocking probability of η_v and a loss probability of γ_v. With admission control, we provision c such that $\bar{m}_v$ VoIP calls may be admitted on the backbone link to achieve the same blocking and loss rates. In general, $\bar{m}_v < \sum_{j=1}^{J} \bar{m}_{v,j}$ because of the statistical multiplexing of connections. However, it is not necessary to perform admission control at the backbone link if $\bar{m}_v = \sum_{j=1}^{J} \bar{m}_{v,j}$ and c is provisioned accordingly. In this case, the backbone link is overprovisioned to account for the worst-case influx of calls.

If the links are all identical, then, for large n, the link must be overprovisioned by a factor of approximately $\frac{(\sqrt{J}-1)Q^{-1}(\eta_v)}{\sqrt{J\lambda_v}+Q^{-1}(\eta_v)}$, where λ_v is the arrival rate of voice calls at each link. (Here, a Gaussian approximation is used to scale $\bar{m}_{v,j}$ and $\bar{m}_v$ with λ_v to achieve a blocking probability of η_v at each link.) For example, if $J = 4$, $\lambda_v = 10$, $\eta_v = 10^{-3}$, then the link must be overprovisioned by 33%. However, if $\lambda_v = 1000$, then this reduces to 13%. Therefore, admission control may be limited to access links provided that the capacity is sufficiently overprovisioned. However, if the overprovisioning required is too steep, then admission control should still be used in the backbone. An attractive alternative is to compute $\bar{m}_v$ on the backbone link using (6), and then to choose $\bar{m}_{v,j}$ with the constraint that $\sum_{j=1}^{J} \bar{m}_{v,j} \leq \bar{m}_v$. These limits could then be modified in a distributed fashion based on actual VoIP usage to minimize the blocking probability.

4.3 Assured Service Provisioning

For Assured Service, the DiffServ nodes must rely on proper provisioning of the AS classes to provide quality to users. With per-class queuing, the performance of TCP depends on the number of active connections for each class, which is typically unknown. However, if the DiffServ network is to provide quality to such users, then the bandwidth allocated to each AS class must be proportional to the current number of active users.

For example, suppose we use two classes of service with WFQ. WFQ assigns bandwidth proportionally to the weight of each backlogged queue. Therefore, we can adjust the bandwidth allocated to each class by assigning a fraction ϕ of the bandwidth to class 1, and $(1-\phi)$ to class 2. Then, the rate per TCP connection, using the link-sharing model of Section 2.1, is $R_1 = \frac{\phi C}{N_1}$ for class 1 and $R_2 = \frac{(1-\phi)C}{N_2}$ for class 2. To provision the link such that $R_1 \approx 2R_2$, we need to set the weight ϕ to be

$$\phi := \frac{2N_1}{2N_1 + N_2}.$$

When N_k is known, the DiffServ nodes may be appropriately provisioned to assure quality. However, often the number of TCP connections must be estimated. In this case, the DiffServ weights must be adjusted based on these estimates.

An important question is the time-scale at which DiffServ weights would need to be recomputed. Using a Gaussian approximation, we see that

$$\frac{2N_1}{2N_1 + N_2} \approx \frac{2\alpha_1}{2\alpha_1 + \alpha_2}\left(1 + \frac{\alpha_2}{2\alpha_1 + \alpha_2}\left(\frac{Z_1}{\sqrt{n\alpha_1}} - \frac{Z_2}{\sqrt{n\alpha_2}}\right)\right).$$

At the access link, with $n = 500$ and $\alpha_1 = \alpha_2 = 10^{-2}$, the relative fluctuation of this ratio has a standard deviation of approximately 21%. Therefore, the ratio exhibits likely fluctuations of up to 35%, and so it is not possible to guarantee the higher quality of class 1 using static weight assignment. This would require the access link to more directly monitor the number of TCP connections that are active. However, within a backbone link, where $n = 50,000$, the standard deviation becomes only 2.1%, so the quality of class 1 is assured. Furthermore, the measurement and estimation in this case can be done at a much slower time-scale; for example, to adjust to a changing user population.

5 Conclusion

In this paper, we introduce a model of TCP behavior at the transaction level, where the number of active TCP connections is a random process. With this model, we demonstrate that Internet quality, in the absence of admission control, can be achieved with statistical multiplexing, but is primarily based on the overprovisioning of links. The degree of overprovisioning largely depends upon the scale of the network.

Furthermore, we extend these models to DiffServ which differentiates between voice calls, using UDP, and data connections, using TCP. Our analysis shows that DiffServ scales well in the backbone, but will require more sophisticated control and monitoring at the access links of the DiffServ network.

References

[1] A. Demers, S. Keshav, and S. Shenker. Analysis and simulation of a fair queuing algorithm. *Journal of Internetworking: Research and Experience*, 1:3–26, 1990.

[2] R. Guerin, H. Ahmadi, and M. Naghshineh. Equivalent capacity and its application to bandwidth allocation in high-speed networks. *IEEE Journal on Selected Areas in Communications*, 9(7):968–981, September 1991.

[3] V. Jacobson. Congestion avoidance and control. *Proceedings of ACM Sigcomm '88, Stanford, CA*, 1988.

[4] M. May, J.-C. Bolot, A. Jean-Marie, and C. Diot. Simple performance models of differentiated services schemes for the Internet. *Proceedings of INFOCOMM '99, New York, NY*, March 1999.

[5] J. Mo. *Fair Congestion Control Protocols for the Internet.* PhD thesis, University of California at Berkeley, 1999.

[6] J. Mo, R. La, J. Walrand, and V. Anantharam. Analysis and comparison of TCP Reno and Vegas. *Proceedings of INFOCOMM '99, New York, NY*, March 1999.

[7] R. Morris. TCP behavior with many flows. *Proceedings of the 1997 International Conference on Network Protocols*, pages 205–211, 1997.

[8] J. Padhye, V. Firoiu, D. Towsley, and J. Kurose. Modeling TCP throughput: A simple model and its empirical evaluation. *Proceedings of SIGCOMM '98, Vancouver, BC, Canada*, September 1998.

Stability of Multilane Input-Buffered Switches with Markov-Modulated Arrival Processes *

Philip Ho, David Tse and Jean Walrand
Department of Electrical Engineering and Computer Sciences
University of California at Berkeley
Berkeley, CA 94720
{pho,dtse,wlr}@eecs.berkeley.edu

Abstract

Implementing multilane buffers at the inputs is a solution to head of line blocking in an input-buffered switch, and it has been proven that under the longest queue first (LQF) scheduling algorithm such switch can achieve 100% throughput. Here we extend this result in two directions: we substitute the assumption of i.i.d. traffic by the more general Markov- modulated traffic; and we establish the result not only for crossbar switches but also banyan networks. The key idea in the proofs is the concept of fluid limit model.

Keywords: ATM switch, banyan network, longest queues first scheduling, fluid limit model, stability

1 Introduction

The performance of high speed ATM networks is, to a large extent, determined by the efficiency of its switching fabrics. Among the different architectures of ATM switches [1], the class of input-buffered switches has the unique advantage that it does not require internal speedup, which becomes more important given the ever growing requirements of bandwidth and degree of multiplexing; while a banyan switch network constructed from switching elements is desirable in its scalability, modularity and self-routing properties. The switch under both of these two categories is the focus of investigation in this paper.

On the flip side, input-buffered switches are notoriously known for their head of line (HOL) blocking phenomenon, which limits the throughput to around 58% [3]. One way to eliminate HOL blocking is to provide separate buffers for cells from the same input that are destined to different outputs, i.e., multilane buffers. In fact McKeown et al. [4] show that using the maximum weight matching, or longest queue first (LQF) scheduling algorithm, an $N \times N$ *crossbar* multilane input-buffered switch can offer 100% throughput. This is the same as saying that the queues are stable. In a different context, Tassiulas et

---

*This work was supported by the NSF Graduate Fellowship.

al. [9] derive that using a modified LQF algorithm, which compares the difference between successive queue lengths along a path, stability can be achieved by a feed-forward data network (which includes the banyan switch). Nevertheless, both algorithms require a centralized control mechanism, and their complexity grows in N^2 which violates the goal of scalability and modularity. Moreover, both results are based on the assumption that the arrival process is i.i.d. and independent among different inputs. This does not quite resemble the bursty nature of real-life ATM traffic. In the following sections we show that there exists a local LQF scheduling algorithm under which a input-buffered banyan network is stable, under a general class of Markov-modulated traffic. It is argued in [8] that this type of process can well describe the burstiness and long-term dependence of real-life traffic.

The notion of stability deserves further discussion. In both works mentioned above the system is modelled as a discrete state Markov chain, and a suitable Lyapunov function is then constructed to establish positive recurrence of the chain, which implies stability. This idea is elaborated in Dai [2], where the author introduces a *fluid limit model* as a tool for proving stability on a wide class of queueing networks: a non-idling queueing system can be shown to converge to its corresponding fluid limit model (FLM) once it is well-defined, a Lyapunov function is then identified with the FLM and hence positive (Harris) recurrence follows. We will follow this methodology in the proofs of our result, with the key difference that in our case the switching network is *not* non-idling, i.e., the server can sit idle even though some of the queues it serves are not empty. For this reason we need to show from first principles that our system indeed converges to its FLM.

The mathematical formulation of a switching element is given in section 2, and the main result is presented and proved in section 3. We conclude in section 4 with our insights and issues of future research.

2 Description of the Switch Model

Although our result is valid for switching elements (SEs) of all sizes (in terms of number of ports), for the sake of simplicity we consider the case of a 2 by 2 multilane input-buffered SE (figure 1), from which a 2^M by 2^M switching network is constructed (figure 2). At each input two queues are maintained, each for one output port. Cells are assumed to be of fixed size, and time is slotted into units of cell time. Let $Q_i(n), \quad i \in G := \{1,2,3,4\}, \quad n = 0,1,2,\ldots$ be the queue lengths at cell time n, as shown in the figure. Two queues are said to be *compatible* if they can receive service at the same cell time. In other words, Q_1 is compatible with Q_4 and Q_2 is compatible with Q_3. Let $\underline{Q}$ be the vector of the queue lengths, $G_1 = G_4 := \{1,4\}$, $G_2 = G_3 := \{2,3\}$ be the sets of compatible queues, and $G_{\bar{j}} = G \backslash G_j$. Let $\sigma_j^Q(n) = \sum_{i\in G_j} Q_i(n)$ and $\sigma_{\bar{j}}^Q(n) = \sum_{i\in G_{\bar{j}}} Q_i(n)$ (the superscript will be dropped when the dependence on queue length process is clear from context). The queues are served under the longest queues first (LQF) policy. Under this policy, the SE serves queues 1 and 4 at time n if $\sigma_1(n) \geq \sigma_2(n)$, otherwise the SE serves queues 2 and 3.

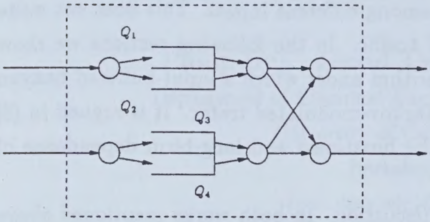


Figure 1: Architecture of a 2 by 2 multilane input-buffered SE

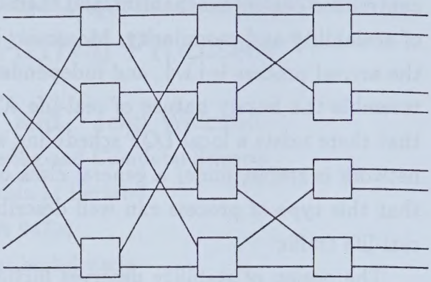


Figure 2: A banyan switch network

Cell arrivals at all inputs are modulated by a discrete time stationary Markov chain $U(n)$ in a finite state space U at each cell time. Let $\underline{\lambda} = [\lambda_i]$ be the rate vector of the stationary arrival process, and $\underline{A}(n)$ be the cumulative arrivals to the queues. We say an arrival process $\underline{A}(n)$ is *admissible* if, for all links inside the switch, the sum of the stationary arrival rates over different input-output pairs that share the link does not exceed the link capacity. For the single SE case this is the same as the capacity constraints at the inputs and outputs: $\lambda_i + \lambda_j < 1$, where $i \in \{1,4\}$, $j \in \{2,3\}$. Throughput this paper we assume the arrival process is admissible.

Since in each queue the cells are served in an FIFO manner. The evolution of the queues can then be expressed as

$$Q_i(n+1) \;=\; [Q_i(n) - W_i^Q(n) + a_i(n+1)]^+ \tag{1}$$

where $W_i^Q(\cdot)$ is the indicator function $\mathbf{1}\{\sigma_i(n) > \sigma_{\bar{i}}(n) \,\text{or}\, G_i \,\text{wins tie breaker}\}$, $[x]^+ = max(x,0)$ and $a_i(n+1) = A_i(n+1) - A_i(n)$. We define ν^1 as the mapping from the initial queue lengths and the arrival process to the queue length processes, i.e.,

$$\nu^1(\underline{Q}(0), \underline{A}(\cdot)) = \underline{Q}(\cdot).$$

We would like to treat $\underline{A}$, $\underline{Q}$ as continuous processes in continuous time in the subsequent proofs (i.e., $\underline{A}$, $\underline{Q} \in C[0,T]$), and that is done using linear interpolation. Moreover, the L_∞ (supremum) norm will be used whenever a metric is required, i.e., $|\underline{Q}(n)| = \max_i Q_i(n)$ and $|\underline{Q}(\cdot)| = \max_{n,i} Q_i(n)$.

At this point we define a Markov chain to describe the dynamics of the system and a mapping from the initial queue lengths and arrivals to the queue length process. The state of any FIFO queue is the output ports of each of the cells in the queue:

$$Y_i(n) := (c_1, c_2, \ldots, c_{Q_i(n)})$$

where $c_q \in$ {set of destinations}. In other words, $Y_i(n)$ is an element in the space of finitely terminated sequences. Such definition is not necessary for a single SE but is needed in the case of banyan networks. Let $\underline{Y}(n) = [Y_i(n)]$ be the states of all queues, then

$$\underline{X}(n) := (\underline{Y}(n), U(n))$$

is the countable state Markov chain that represents the dynamics of the system. We will sometimes use $\underline{X}^{\underline{z}}(n)$, where $\underline{z} := \underline{Q}(0)$, to indicate the chain's dependence on initial queue lengths. A pseudo-norm for $\underline{X}(n)$ can be defined as follows:

$$|\underline{X}(n)| := |\underline{Q}(n)|.$$

Note that the level sets $\{\underline{x} \in U : |\underline{x}| \leq K\}$, $K \in R^+$ are compact.

3 Main Theorem

The main result of the paper is the following theorem:

Theorem 1 (Positive Recurrence) *The Markov chain $\{\underline{X}(n), n \geq 0\}$ is positive recurrent. Moreover, the same is true for the Markov chain that models the state of the banyan network of switching elements.*

The key steps of the proof are as follows:

1. We define a fluid limit model. The interpretation of this model is that the cell time is assumed to be very small so that the cell streams look like fluids and the queue occupancies can be approximated by continuous state processes. This fluid limit is deterministic and the arrivals have the average rates. Moreover, it converges to an empty system in finite time if the arrivals are admissible. (Section 3.1)
2. We apply a proper scaling to the state process of the system and prove that the scaled process indeed converges (in L_∞) to the FLM. (Section 3.2)
3. We then show that convergence to zero queue lengths in finite time in the fluid limit model implies positive recurrence of the state process. This fact is based on a key result from [2]:

   Theorem 2 ([2],Thm. 3.1) *If there exists $\delta > 0$ such that $|\underline{q}|\delta \in Z^+$ and*

   $$\lim_{|\underline{q}|\to\infty} \frac{1}{|\underline{q}|} E|\underline{X}^{\underline{q}}(|\underline{q}|\delta)| = 0$$

   then $\underline{X}(\cdot)$ is positive Harris recurrent.

We will show that the premise of this theorem follows from the statements of 1 and 2. Note that positive Harris recurrence is equivalent to positive recurrence in a countable state space, which is our notion of switch stability. (Section 3.3)

4. Finally we extend the result to the switching network. In particular, we would validate the previous statements in a banyan network setting and establish positive recurrence of the Markov chain that describes the dynamics of the network, given that each SE runs the LQF algorithm locally. The main argument for this part is induction on the stages of the banyan switch. (Section 3.4)

3.1 The Fluid Limit Model

To characterize the FLM we need the notion of a continuous LQF (CLQF) policy. Intuitively it is just the LQF policy executed continuously: for some arrival process $\underline{A}$ and initial queues $\underline{q}(0)$, when $\sigma_j(t) > \sigma_{\bar{j}}(t)$,

$$\dot{q}_i(t) = \begin{cases} (\dot{A}_i(t) - 1)\mathbf{1}\{q_i(t) = 0\} & : \; i \in G_j \\ \dot{A}_i(t) & : \; i \in G_{\bar{j}} \end{cases}$$

whenever $\underline{\dot{A}}(t)$ exists; and because of its continuity $\underline{q}$ is well defined. Note that $\dot{\sigma}_j(t) = \sum_{i \in G_j} \dot{q}_i(t)$ is strictly decreasing while $\dot{\sigma}_{\bar{j}}(t)$ is non-decreasing, therefore there exists T_0 such that $\sigma_j(T_0) = \sigma_{\bar{j}}(T_0)$. Since CLQF aims to maintain this equality, the system will serve the queues such that

$$\dot{\sigma}_j(t) = \dot{\sigma}_{\bar{j}}(t) \tag{2}$$

for $t \geq T_0$. In other words,

$$\sum_{i \in G_1} (\dot{A}_i(t) - s_i(t))\mathbf{1}\{q_i(t) > 0\} = \sum_{i \in G_2} (\dot{A}_i(t) - s_i(t))\mathbf{1}\{q_i(t) > 0\} =: u(t). \tag{3}$$

Note that $u(t)$ is the common rate of change of the queue sums, and $s_j(t)$ is the rate at which the SE is serving G_j, and that $s_1(t) + s_2(t) = 1$. We can now define ν as the mapping for CLQF, i.e., $\nu(\underline{q}(0), \underline{A}(\cdot)) = \underline{q}(\cdot)$, and the FLM $\underline{\tilde{Q}}(t)$ is simply the mapping of $(\underline{z}, \underline{\lambda} t)$ under ν, where $|\underline{z}| = 1$. Our first lemma asserts that the FLM under admissible arrivals empties itself in finite time:

Lemma 1 *There exists $\delta_0 < \infty$ such that $\underline{\tilde{Q}}(t) = 0$ for all $t > \delta_0$. In fact, this is true for all $\nu(\underline{z}, \underline{\lambda} t)$ such that $|\underline{z}|$ is bounded and $\underline{\lambda}$ is admissible.*

Proof. Note that $\dot{A}_i(t) = \lambda_i$, and without loss of generality consider the FLM at time T_0 as defined above. It is clear that $|\underline{\tilde{Q}}(T_0)|$ is bounded, and admissibility constraint implies that there exist $s_1(t)$ and $s_2(t) = 1 - s_1(t)$ such that $u(t)$ is negative and bounded away from zero. Hence $\sigma_j(t)$, $j = 1, 2$ are strictly decreasing, and thus reaches zero within a finite time δ_0, after which all queues remain empty. ■

3.2 Convergence to the Fluid Limit Model

We now define the k-th order rescaled system. Let

$$\underline{A}^k(t) := \tfrac{1}{k}\underline{A}(kt) \quad \underline{Q}^k(t) := \tfrac{1}{k}\underline{Q}(kt);$$

the corresponding rescaled mapping ν^k is such that

$$\nu^k(\underline{Q}^k(0), \underline{A}^k(\cdot)) := \underline{Q}^k(\cdot)$$

whenever $\nu^1(\underline{Q}(0), \underline{A}(\cdot)) = \underline{Q}(\cdot)$. In other words, the rescaled system serves 1/k units of cells every 1/k cell time under the LQF policy. We can now establish the convergence of the rescaled process to the FLM:

Lemma 2 *For any arrival process $\underline{A}(t)$ modulated by a finite state Markov chain with stationary arrival rate $\underline{\lambda}$ that is admissible, and $\underline{Q}^k(0) \to \underline{z}$, we have $\underline{Q}^k(t) \to \tilde{\underline{Q}}(t)$.*

To show this result we need two more lemmas:

Lemma 3 *ν^k is uniformly continuous for each $k \in Z^+$.*

Lemma 4 *ν^k converges to ν uniformly on compact sets.*

Proof. See appendix for the proofs of the lammas 2, 3 and 4. ■

3.3 Positive Recurrence of Switch State Process

We now argue that the condition in Theorem 2 holds:

Lemma 5 *There exists $\delta > 0$ such that*

$$\lim_{|\underline{q}|\to\infty} \frac{1}{|\underline{q}|} E|\underline{X}^{\underline{q}}(|\underline{q}|\delta)| = 0$$

Proof. Note that $H := \{\underline{r} : |\underline{r}| = 1\}$ is a compact set, hence the result in lemma 1 is uniform over H, i.e., for a fixed arrival process there exists $\delta = \max_{r \in H}\{\delta_0\}$ such that $|\tilde{\underline{Q}}(t)| = 0$ for $t > \delta$ uniformly over initial queue lengths in H. Hence for any $\epsilon > 0$ there exist K such that for $k := |\underline{Q}(0)| > K$,

$$\frac{1}{k} E|\underline{X}^{\underline{q}}(\lceil k\delta \rceil)| = E|\underline{Q}^k(\delta')| = E|\underline{Q}^k(\delta') - \tilde{\underline{Q}}(\delta')| < \epsilon$$

where we take $\underline{q} = \underline{Q}(0)/k$, $\delta' = \lceil k\delta \rceil / k \geq \delta$, and $\tilde{\underline{Q}}(t) = \nu(\underline{q}, \underline{\lambda} t)$ such that $\tilde{\underline{Q}}(\delta') = 0$. ■

3.4 Extension to Banyan Networks

We would redefine our variables for the case of a banyan switch network with M stages. Let $\underline{Q}_m(n)$ be the vector of queue lengths at stage m with supremum norm, and $\underline{Y}_m(n)$ be the corresponding queue states. Let $\underline{Q}_{net} := (\underline{Q}_1, \ldots, \underline{Q}_M)$ and $\underline{Y}_{net} := (\underline{Y}_1, \ldots, \underline{Y}_M)$. Let $\underline{A}_1(n)$ be the Markov-modulated external arrivals to the banyan network with mean $\underline{\lambda}$ which honors the admissibility constraints, and $\underline{A}_{m+1}(n) := \underline{A}_m(n) + \underline{Q}_m(0) - \underline{Q}_m(n)$ be the arrivals to stage $m+1$ (which is the departure from the stage m.) Let $\mu^k(\underline{Q}_{net}(0), \underline{A}_1(\cdot))$ and $\mu(\underline{Q}_{net}(0), \underline{A}_1(\cdot))$ be the k-th order rescaled LQF system the CLQF system respectively, and $\tilde{\underline{Q}}_{net}(t) := \mu(\tilde{\underline{Q}}_{net}(0), \underline{\lambda} t)$ be the fluid limit model. We also modify ν^k and ν such that they map all the inputs and the initial queue lengths in a particular stage in the banyan network to the queue length processes of that stage in their respective systems. It is easy to see that the properties of ν^k and ν that we have proved stay true under this new definition, for at each stage we have an array of SEs that operate independently. Finally let $\underline{X}_{net}(\cdot) := (\underline{Y}_{net}(\cdot), U(\cdot))$ be the state of the system. To prove the second part of theorem 1 we need the show lemmas 1 through 5 remain valid in the network setting. Let us refer to them as corollaries 1 through 5. It turns out that corollaries 1 and 4 can be proved via induction on the stages m:

Corollary 1 *There exists* $\delta_M < \infty$ *such that* $\tilde{\underline{Q}}_{net}(t) = \underline{0}$ *for all* $t > \delta_M$.

Corollary 4 μ^k *converges uniformly to* μ.

Proof. See appendix for the proofs of corollaries 1 and 4. ■

The continuity of μ^k (corollary 3) is proved in the exact same way as the single SE case, and with corollary 4 it implies convergence to the FLM (corollary 2). Positive recurrence in the network setting is then established as the condition of theorem 2 are fulfilled and the theorem is proved.

4 Conclusion and Future Work

We have shown that, with the aid of a fluid limit model, a banyan network of multilane input-buffered crossbar switching elements can achieve 100% throughput under Markov-modulated input traffic if the arrivals are admissible and the long queues first scheduling algorithm is executed locally within each SE. It is worth noting that the model assumes arbitrary large buffers are allocated to each SE in the network, which enables its capacity region to be defined only by the link constraints. On the other hand, the capacity region of a banyan network where buffers are implemented only at the first stage is also constrained by the convex hull defined by the configurations of the SEs (parallel vs. cross forwarding). It would be interesting to determine the relationship between the size and position of the buffers and the capacity region under various scheduling policies. Moreover, recent works (e.g. see [6]) have also demonstrated the use of fluid limit models in optimal control of queueing networks. This provides a powerful tool to investigate the efficiency of different scheduling policies for more sophisticated switching architectures.

References

[1] R.Y. Awdeh & H.T. Mouftah: "Survey of ATM Switch Architectures", Comp. Net. & ISDN Sys., vol.27, p.1567-1613.

[2] J. Dai: "On Positive Harris Recurrence of Multiclass Queueing Networks: A Unified Approach Via Fluid Limit Models", Annals of App. Prob., vol.5, p.49-77.

[3] M. Karol,M.G. Hluchyj & S.P. Morgan: "Input Versus Output Queueing on a Space-Division Packet Switch", IEEE Tran. Comm., vol.COM-35(12), p.1347-1356.

[4] N. McKeown,V. Anantharam & J. Walrand: "Achieving 100an Input-Queued Switch", Proc. IEEE infocom '96.

[5] N. McKeown: "Scheduling Algorithms for Input-Queued Cell Switches", PhD Thesis, UCB, May 1995.

[6] S.P. Meyn: "Stability and Optimization of Queueing Networks and their Fluid Models", Proc. "The Mathematics of Stochastic Manufacturing Systems", Williamsburg, VA, '96.

[7] S.P. Meyn & R.L. Tweedie: "Markov Chains & Stochastic Stability", Springer-Verlag, '93.

[8] S. Robert & J.Y. LeBoudec: "On a Markov Modulated Chain Exhibiting Self-Similarities Over Finite Timescale", Perf. Eval., vol.27-28, p.159-173.

[9] L. Tassiulas & A. Ephremides: "Stability Properties of Constrained Queueing Systems and Scheduling Policies for Maximum Throughput in Multihop Radio Networks", IEEE Tran. Auto. Ctrl., vol.37(12), p.1936-1948.

Appendix: Proofs

Proof of Lemma 3. Consider $(\underline{Z}(0), \underline{B}(\cdot))$ and $(\underline{\hat{Z}}(0), \underline{\hat{B}}(\cdot))$. We first show by induction that for any $\epsilon > 0$, there exists δ_n such that if $\underline{Z}(\cdot) = \nu^k(\underline{Z}(0), \underline{B}(\cdot))$, $\underline{\hat{Z}}(\cdot) = \nu^k(\underline{\hat{Z}}(0), \underline{\hat{B}}(\cdot))$ then

$$|\underline{Z}(n) - \underline{\hat{Z}}(n)| < \epsilon \tag{4}$$

whenever $|\underline{Z}(0) - \underline{\hat{Z}}(0)| + |\underline{B}(n) - \underline{\hat{B}}(n)| < \delta_n$, for $n = 0, 1/k, \ldots, T$. Taking $\delta_0 = \epsilon$ and the statement is true for $n = 0$. Now choose $\epsilon' < \epsilon/3$ such that $\sigma_1^Z(n) - \sigma_2^Z(n)$ and $\sigma_1^{\hat{Z}}(n) - \sigma_2^{\hat{Z}}(n)$ have the same sign, and by induction hypothesis we can choose δ_n such that (4) holds for ϵ'. Now

$$\begin{aligned}
&\left|\underline{Z}(n+\tfrac{1}{k}) - \underline{\hat{Z}}(n+\tfrac{1}{k})\right| \\
&\leq |\underline{Z}(n) - \underline{\hat{Z}}(n)| + |\underline{W}^{Z(0)}(n) - \underline{W}^{\hat{Z}(0)}(n)| + |\underline{B}(n+\tfrac{1}{k}) - \underline{\hat{B}}(n+\tfrac{1}{k})| + |\underline{B}(n) - \underline{\hat{B}}(n)| \\
&< \tfrac{\epsilon}{3} + 0 + \tfrac{2\epsilon}{3} = \epsilon
\end{aligned}$$

if we take $\delta_{n+1/k} = \min(\delta_n, \epsilon/3)$. Taking $\delta = \min\{\delta_n\} = \delta_T > 0$ we have (4) holds for $n = 0, 1/k, \ldots, T$, and uniform continuity follows since the other points of the processes are obtained by linear interpolation, and the choice of δ_n depends only on the difference between the inputs. ∎

Proof of Lemma 4. Consider $\underline{Z}^k(\cdot) = \nu^k(\underline{Z}(0), \underline{B}(\cdot))$ and $\underline{Z}(\cdot) = \nu(\underline{Z}(0), \underline{B}(\cdot))$. We would like to establish the closeness of $\underline{Z}^k$ and $\underline{Z}$. Note that as long as $n \leq T_0$ the dynamics of the two systems are identical. Hence we can assume that the systems start at $n_0 = \lfloor kT_0 \rfloor / k$. Moreover, let $\psi_i(t) = Z_i^k(t) - Z_i(t)$.

We first need the fact that

$$|\sum Z_i^k(n) - \sum Z_i(n)| \leq \frac{C}{k} \tag{5}$$

for $n = 0, 1/k, \ldots, T$. This is merely saying that the difference in total work done for the two systems are bounded in the order of $1/k$. Now using a result from ([5],Thm. 4.3) that $|\sigma_1^{Z^k}(n) - \sigma_2^{Z^k}(n)| \leq 4/k$, and the definition that $\sigma_1^Z(n) = \sigma_2^Z(n)$, we have

$$|\sigma_1^{Z^k}(n) - \sigma_1^Z(n)| = |\psi_1(n) + \psi_4(n)| \leq \frac{1}{k}\left(\frac{C}{2} + 2\right) =: \frac{C'}{k}. \tag{6}$$

We argue that if $\psi_1(n)$ and $\psi_4(n)$ have opposite signs (say $\psi_1(n) > 0$ and $\psi_4(n) < 0$ for $n \in [a, b]$) then $|\psi_1(n) - \psi_4(n)|$ is non-increasing. Together with (6) this would imply $|\psi_1(n) - \psi_4(n)| \leq C'/k$ for all n, and thus $|\psi_1(n)|, |\psi_4(n)| \leq C'/k$ and our desired result follows. (The case is identical for $\psi_2(n)$ and $\psi_3(n)$.) Note that $\psi_1(n)$ and $\psi_4(n)$ are either both increasing or both decreasing, and the magnitude of their changes are the same unless one of the queues becomes empty. Suppose $\psi_1(n)$, $\psi_4(n)$ are decreasing, then $Z_1^k(n) \geq \psi_1(n) > 0$ means $Z_1^k(n)$ may not be empty until $\psi_1(n) \leq 0$, i.e., when $n > b$. This implies $\psi_1(n) - \psi_1(m) \geq \psi_4(n) - \psi_4(m)$ and therefore

$$|\psi_1(m) - \psi_4(m)| = \psi_1(m) - \psi_4(m) \leq \psi_1(n) - \psi_4(n) = |\psi_1(n) - \psi_4(n)|$$

for $m \in [n, b]$. ∎

Proof of Lemma 2. We first show that $\underline{A}^k(t) \to \underline{\lambda} t$. For any $\delta > 0$, when $t < \delta$, $|\underline{A}^k(t) - \underline{\lambda} t| < \max\{|\underline{A}^k(t)|, |\underline{\lambda} t|\} < \delta$. When $\delta \leq t \leq T$ let $\delta' = \frac{\delta}{T} > 0$, and the ergodic theorem states that there exists $K < \infty$ such that

$$\left|\frac{\underline{A}(kt)}{kt} - \underline{\lambda}\right| < \delta'$$

in L_1 for $kt > K$, uniformly over (finite) initial states. This implies

$$\left|\frac{\underline{A}(kt)}{k} - \underline{\lambda} t\right| = \left|\underline{A}^k(t) - \underline{\lambda} t\right| < \delta' t < \delta$$

Now

$$\begin{aligned} \left|\underline{Q}^k(t) - \underline{\tilde{Q}}(t)\right| &= \left|\nu^k(\underline{Q}^k(0), \underline{A}^k(\cdot)) - \underline{\tilde{Q}}(t)\right| \\ &\leq \left|\nu^k(\underline{Q}^k(0), \underline{A}^k(\cdot)) - \nu^l(\underline{Q}^k(0), \underline{A}^k(\cdot))\right| + \left|\nu^l(\underline{Q}^k(0), \underline{A}^k(\cdot)) - \nu^l(\underline{\tilde{Q}}_0, \underline{\lambda} t)\right| \\ &\quad + \left|\nu^l(\underline{q}, \underline{\lambda} t) - \underline{\tilde{Q}}(t)\right| \end{aligned}$$

for some sufficiently small l. The first and third terms are bounded by ψ because of uniform convergence of ν^k to ν (lemma 3), and so is the second term which follows from the continuity of ν^l (lemma 4). This shows the convergence of $\underline{Q}^k(t)$ to $\underline{\tilde{Q}}(t)$. ■

Proof of Corollary 1. From lemma 1 each SE in stage 1 can empty its queue in finite time, hence there exists $\delta_1 < \infty$ such that $\underline{Q}_1(t) = \underline{0}$ for all $t > \delta_1$. By induction hypothesis, $|\underline{Q}_{m+1}(\delta_m)| < \infty$ at time δ_m and $\underline{A}_{m+1}$ is equal to its respective admissible mean rate. Then by lemma 1 there exists $\beta_{m+1} = \max\{\delta_0\} < \infty$ over all SEs at stage $m+1$, such that $\underline{Q}_{m+1}(t) = \underline{0}$ for all $t > \delta_m + \beta_{m+1} =: \delta_{m+1}$. ■

Proof of Corollary 4. let $\underline{Z}^k_{net} = \mu^k(\underline{Y}_{net}, \underline{B}_1)$ and $\underline{Z}_{net} = \mu(\underline{Y}_{net}, \underline{B}_1)$. From lemma 4, for $\epsilon_1 > 0$ there exists k such that $|\underline{Z}^k_1 - \underline{Z}_1| < \epsilon_1$. Moreover, $|\underline{B}^k_1 - \underline{B}_1| = 0 < \alpha_1$. The induction hypothesis states that $|\underline{Z}^k_m - \underline{Z}_m| < \epsilon_m$ and $|\underline{B}^k_m - \underline{B}_m| < \alpha_m$. Now

$$\begin{aligned} |\underline{B}^k_{m+1} - \underline{B}_{m+1}| &= |(\underline{B}_m + \underline{Y}_m - \underline{Z}^k_m) - (\underline{B}_m + \underline{Y}_m - \underline{Z}_m)| \\ &\leq |\underline{B}^k_m - \underline{B}_m| + |\underline{Z}^k_m - \underline{Z}_m| \\ &< \alpha_m + \epsilon_m =: \alpha_{m+1}. \end{aligned}$$

and

$$\begin{aligned} |\underline{Z}^k_{m+1} - \underline{Z}_{m+1}| &= |\nu^k(\underline{Y}_{m+1}, \underline{B}^k_{m+1}) - \nu(\underline{Y}_{m+1}, \underline{B}_{m+1})| \\ &\leq |\nu^k(\underline{Y}_{m+1}, \underline{B}^k_{m+1}) - \nu^k(\underline{Y}_{m+1}, \underline{B}_{m+1})| \\ &\quad + |\nu^k(\underline{Y}_{m+1}, \underline{B}_{m+1}) - \nu(\underline{Y}_{m+1}, \underline{B}_{m+1})| \\ &< \beta_{m+1} + \gamma_{m+1} =: \epsilon_{m+1}. \end{aligned}$$

Note that β_m depends on α_m, and γ_m depends on k. For $\epsilon > 0$, let $\epsilon_M = \epsilon$, and α_M, k_M be such that β_M, $\gamma_M < \epsilon_M/2$. Moreover, for $m = 1, \ldots, M-1$, let $\epsilon_m = \min\{\alpha_{m+1}/2, \epsilon\}$, k_m be such that $\gamma_m < \epsilon_m/2$, and α_m be such that $\beta_M < \epsilon_M/2$ or $\alpha_{m+1}/2$, whichever is smaller. Then taking $k = \max\{k_m\}$ we have

$$|\underline{Z}^k_{net} - \underline{Z}_{net}| < \epsilon$$

and uniform convergence follows. ■

Adaptive Transmission for Spread-Spectrum Communications over Multipath Channels

Michael B. Pursley and Clint S. Wilkins
Department of Electrical and Computer Engineering, Clemson University
337 Fluor Daniel Building
Clemson, SC 29634

Abstract

An adaptive transmission system for direct-sequence spread-spectrum packet communication networks is proposed for use on multipath channels to provide energy-efficient communication in a time-varying environment. The power, code rate, and symbol rate are the transmission parameters that are adapted to match the path loss and multipath profile of the channel. The adaptation is based on certain statistics that are derived in the spread-spectrum receiver. The optimization criterion used in the selection of transmission parameters in the adaptive system is related to the energy consumption and time duration of a packet transmission.

I. Introduction

The use of adaptive transmission is investigated for direct-sequence (DS) spread-spectrum communications. The transmission parameters that are adapted are the power of the transmitted signal, the rate of the error-control code, and the symbol rate. The goal of an adaptive transmission system is to match the transmission parameters to the channel in order to provide the prescribed error probability while keeping the energy consumption and transmission time as small as possible. A reduction in the energy consumption prolongs battery lifetime in mobile terminals, and a decrease in the transmission time for each packet improves the network throughput.

The selection of the transmission parameters in the proposed adaptive transmission system is based on a criterion that accounts for both energy consumption and transmission time. Using this criterion, we determine the optimum combination of transmission parameters for each of several multipath profiles. It is desirable for the transmission system to employ the optimum parameters for each packet transmission, but the channel varies with time and the transmitter does not know the channel state. As a result, the adaptive transmission system must rely on feedback from the receiver to the transmitter. Even the receiver does not know the channel state, but it can make measurements that provide some information about the state of the channel. In our approach, the adaptation of the transmission parameters is based on certain statistics that are determined in the receiver and sent to the transmitter in acknowledgement packets. This information can also be included in a clear-to-send (CTS) message if the channel-access protocol employs an exchange of request-to-send (RTS) and CTS messages before a data packet is transmitted.

We have previously investigated adaptive transmission for frequency-hop (FH) spread-spectrum systems and networks. Methods for the adaptation of the code rate in FH systems are described and analyzed in [1], and code-rate and transmitter-power adapta-

---

This research was supported in part by the Army Research Office under grant DAAH04-95-1-0247, the Office of Naval Research under grant N00014-97-1-0776, and the Defense Advanced Research Projects Agency under contract DAAB07-97-C-D012.

tion techniques are proposed and analyzed in [2]. Adaptive transmission protocols for FH spread-spectrum networks are evaluated in [3], and adaptive transmission protocols and adaptive routing algorithms are integrated to provide energy-efficient transmission and routing in [4]. The adaptive transmission protocols that we have proposed for FH spread-spectrum systems and networks do not require estimates of the power in the received signal. Instead, the adaptation is based only on counts of errors and erasures in the FH receiver. In contrast, the adaptive transmission system proposed in this paper for DS spread-spectrum systems does require the receiver to make certain measurements of received signal power. In addition, erasures are not usually made in a DS spread-spectrum receiver, so only an estimate of the number of errors is available to the DS spread-spectrum adaptive transmission system.

II. System and Channel Models

The DS spread-spectrum system employs binary differential phase-shift key modulation and rake reception with equal-gain noncoherent combining. The modulation and combining methods are adopted primarily because they are relatively easy to implement (e.g., compared with coherent demodulation and optimal combining). The rake receiver has four taps, so contributions from up to four multipath components can be combined. Convolutional coding is employed with block interleaving and soft-decision Viterbi decoding.

The spreading sequence is modeled as a random binary sequence with a period that is much longer than the duration of a single transmission. The chip rate is held constant as the code rate and symbol rate are varied. The duration of a binary channel symbol is the chip duration times the spreading factor (which is the number of chips per channel symbol). Because the chip duration is held constant, the spreading factor is proportional to the symbol duration. The number of information bits per packet is also held constant as the code rate and symbol rate are varied. For the numerical results, the number of information bits per packet is 1000.

The spreading factors available to the adaptive system are 8, 16, 32, and 64 chips per channel symbol, and the available rates for the binary convolutional code are 1/2, 3/4, and 1 (i.e., no coding). The constraint length of the rate 1/2 code is seven and the minimum free distance is ten [5]. The rate 3/4 code is formed by puncturing the rate 1/2 code, and this permits the use of the same decoder for the two codes.

The focus in this paper is on the effects of multipath, and it is assumed that multiple-access interference is negligible. This is consistent with the use of a reservation-based channel-access technique, such as one that relies on the RTS/CTS exchange mentioned above. If such a channel-access technique operates as intended, there is no multiple-access interference in the data communication channel. The channel is modeled as a specular multipath channel. This is a valid model if the individual multipath components are fully resolved by the rake receiver, which is appropriate for DS spread-spectrum systems with a sufficiently high chip rate.

III. Optimum Transmission Parameters

Two communications resources that should be conserved are energy and time. By *energy* we mean the amount of energy consumed by the transmitter in the transmission of a packet. By *time* we mean the amount of time that is required in the transmission of the packet. The desire to minimize energy needs no further explanation, but the concept of transmission time as a communications resource is not as widely appreciated. The transmission time, which is also referred to as the *on-air time*, should be kept to a minimum, because a long transmission time causes a long busy period for both the transmitting and receiving terminals. During the period that these two terminals are busy with the transmission and reception of a packet, they are unavailable to handle other traffic in the network.

We must account for the fact that the probability of error is not determined by the energy alone; instead, it is a function of E_b/N_0, the ratio of the energy per information bit to the one-sided spectral density of the thermal noise. An obvious way to permit a reduction in energy with no loss in performance is to reduce the spectral density of the thermal noise by decreasing the noise figure of the receiver. But this increases the cost and complexity of the communication terminal, and so reducing energy requirements by decreasing the noise figure below a reasonable level does not yield a net reduction in the cost and complexity of the communication terminals. One way to account for the tradeoff between the energy and the spectral density of the thermal noise is to normalize the energy by dividing it by the noise density. With this in mind we adopt the energy-to-noise-density ratio as the measure of energy efficiency. The energy required to transmit a packet is the number of information bits per packet times the energy per information bit. Throughout this paper, E_b is the total energy per bit in all of the multipath components. Although the energy consumption for a packet transmission is the product of E_b and the number of bits per packet, only E_b varies because the number of information bits per packet is held constant as the transmission parameters are changed. Thus E_b/N_0 is an appropriate measure of energy efficiency for the adaptive transmission system.

The time required to transmit a packet is the product of the number of bits per packet, the chip duration, and the quantity N/r, where r is the rate of the error-control code and N is the spreading factor (i.e., the number of chips per symbol). The number of bits per packet and the chip duration are held constant as the transmission parameters are changed, so the on-air time for the adaptive transmission system is proportional to N/r. Thus N/r is an appropriate measure of transmission time for the adaptive transmission system.

We would like for the adaptive transmission system to minimize the energy-to-noise-density and minimize the on-air time for each packet transmission, but these are conflicting goals in many situations. As a result, and in order to simplify the optimization problem, we introduce a single parameter that is designed to represent a combination of the energy consumption and the transmission time. This parameter, referred to as the *energy-time* product and denoted by W, is proportional to the product of the energy-to-noise density and the time required to transmit the packet. Specifically, we define the energy-time product by $W = (N/r)(E_b/N_0)$. Note that W is a function of all three transmission parameters: the transmitter power, the code rate, and the symbol rate. Use of the energy-time product provides a single parameter to serve as the optimization criterion.

For a given channel, the *optimum transmission parameters* are the power, code rate, and symbol rate that give the minimum value of the energy-time product subject to a constraint on the packet error probability. For the numerical results presented in this paper, we require that the packet error probability not exceed 10^{-2}. In Tables 1-3 the required value of E_b/N_0 (in dB) and the packet transmission time T_p are given for different spreading factors and code rates for three different multipath channels. Transmission times are presented in terms of T, which is the time required to transmit an uncoded packet using $N = 8$. For the entry that is blank, the required probability of error cannot be achieved for any value of E_b/N_0. This situation does not arise if there is no multipath, but the intersymbol interference on a multipath channel may cause errors even if there is no thermal noise. If on such a channel the probability of error for noise-free reception exceeds the required probability of error, the required probability of error

	N=8
r=1/2	6.8, $2T$
r=3/4	7.0, $4T/3$
r=1	10.3, T

Table 1. Energy and transmission time, $(E_b/N_0, T_p)$ for a channel with no multipath.

	N=8	N=16	N=32	N=64
r=1/2	8.4, $2T$	7.8, $4T$	7.7, $8T$	7.6, $16T$
r=3/4	9.4, $4T/3$	8.2, $8T/3$	7.8, $16T/3$	7.7, $32T/3$
r=1	> 25, T	13.9, $2T$	11.8, $4T$	11.1, $8T$

Table 2. Energy and transmission time, $(E_b/N_0, T_p)$, for a channel with two equal-strength components.

cannot be achieved no matter how much energy is transmitted. In effect, an increase in the transmitted energy increases both the desired signal and the interference.

The multipath channels are characterized by the number of multipath components that are received and their relative time delays. The multipath components are assumed to be of equal strength for the results in Tables 1-3, but the numerical values in these tables are not very sensitive to the relative strengths of the multipath components. Furthermore, for a given total energy in the received signal, a uniform distribution of energy among the multipath components gives a larger probability of error than any nonuniform distribution. The difference in the time delays for the multipath components in the two-component channel is six chips, and the multipath components in the four-component channel are equally spaced at intervals of six chips. Relative time delays that are integer multiples of the chip duration give approximately the maximum error probability over all time delays. So the values of E_b/N_0 in the tables are sufficient regardless of the distribution of energy among the multipath components or their relative time delays.

The optimum transmission parameters are determined from the results in Tables 1-3. These are summarized in Table 4, where the results for a three-component multipath channel are also included. If the adaptive transmission system were given perfect information about the channel, it would be a simple matter to select the optimum transmission parameters from Table 4. In practice, however, the adaptive transmission system must estimate certain parameters and then use these estimates in conjunction with Table 4 to select the transmission parameters.

IV. Adaptive Transmission Algorithm

The first step in the proposed adaptive transmission algorithm is to estimate the value of E_b/N_0 on the link over which the transmission is to be made. The information required to estimate the value of E_b/N_0 is provided by certain statistics that are determined from the most recent reception on that link. These statistics are described in the next section.

Next the adaptive transmission system estimates the number of multipath components that were present in the previous reception on the link. The information required to estimate the number of multipath components is provided by the rake receiver. If the

	N=8	N=16	N=32	N=64
r=1/2	11.2, $2T$	9.2, $4T$	8.7, $8T$	8.5, $16T$
r=3/4	20.1,$4T/3$	9.9, $8T/3$	8.9, $16T/3$	8.6, $32T/3$
r=1	—	20.0, $2T$	12.5, $4T$	12.1, $8T$

Table 3. Energy and transmission time, $(E_b/N_0, T_p)$, for a channel with four equal-strength components.

Number of components	Spreading factor	Code rate	E_b/N_0 (target)
1	8	3/4	7.0 dB
2	8	3/4	9.4 dB
3	8	1/2	9.8 dB
4	8	1/2	11.2 dB

Table 4. Optimum transmission parameters for four multipath profiles.

rake receiver is working perfectly and if the number of multipath components is less than the number of correlators in the rake receiver, the number of multipath components is known exactly. Note that if all correlators are active, there is no way to know if the number of multipath components is greater than the number of correlators. For this reason, and because some multipath components may not be acquired even if the number of components is less than the number of correlators, the number of active correlators is not necessarily equal to the number of multipath components. However, the number of active correlators is a lower bound on the number of multipath components, and we use this lower bound as the estimate of the number of multipath components in our adaptive transmission system.

The statistics used to estimate E_b/N_0 and the number of active correlators are determined in the receiver. The receiver provides the necessary information to the transmitter in a feedback message that is sent in response to the most recent reception on that link. As mentioned in the Introduction, the feedback mechanism can be an acknowledgment packet, an RTS/CTS exchange, or a combination of these.

Based on estimates of E_b/N_0 and the number of multipath components, the optimum transmission parameters are determined from Table 4. As an example, suppose that on the most recent transmission over this link it is estimated that there are four multipath components and that E_b/N_0 is 6.0 dB. According to Table 4, the next transmission should use eight chips per channel symbol (i.e., a symbol rate that is one-eighth of the chip rate), a code of rate 1/2, and $E_b/N_0 = 11.2$ dB. This means that the transmitter power should be 5.2 dB larger than it was for the previous transmission on this link. In practice one may want to include some margin to account for measurement errors and changes in the path loss since the last transmission on the link, in which case the increase in transmitter power might be 5.5 dB to 6.5 dB, or even higher if the uncertainties are large.

V. Receiver Statistics

Let the total power in the received signal, including all multipath components, be denoted by S. The *signal-to-noise-density ratio*, denoted by S/N_0, is equal to the product of E_b/N_0 and the information rate. The value of S/N_0 provides the same information to the transmitter as the value of E_b/N_0, since the information rate is known to the transmitter. When expressed in decibels, the units of S/N_0 are dB-Hz.

The numerical values for the receiver statistics that we describe are plotted in Figures 1-3 as a function of S/N_0 in order to make the curves independent of the information rate. The curves in each of these figures correspond to three different multipath profiles. One curve is for a single signal component (i.e., no multipath), one is for a multipath channel that generates two equal-strength multipath components, and one curve is for a multipath channel that produces four equal-strength multipath components

Certain statistics can be computed in the receiver as a packet is being received, demodulated, and decoded. One output of the automatic gain control (AGC) system in the receiver can be used to obtain an estimate of the total power in the signal that is present in the front end of the receiver. This signal consists of the sum of the thermal noise and all of the multipath components. Because the combination of noise and signal components is a random process, the AGC statistic is a random variable. Typical plots of the mean of the AGC statistic as a function of the signal-to-noise-density ratio are shown in Figure 1 for three multipath profiles. Although the mean of the AGC statistic depends on the signal power and the noise density individually (not just on the ratio S/N_0), this is of no importance if the noise density is constant. If there are significant variations in the noise figure of the receiver, periodic calibrations of the AGC system may be required.

The AGC statistic is a pre-detection statistic in the sense that it is derived from the received signal prior to demodulation. There are several types of post-detection statistics that can be obtained at the outputs of the correlators in the receiver. These post-detection statistics are often more reliable because of the beneficial effects of the correlation process applied to the noise-corrupted signal. We have evaluated one post-detection statistic, which we refer to as the *post-detection signal quality* (PDSQ) statistic. The PDSQ statistic is specific to a particular hardware implementation, and a full description of the implementation is beyond the scope of this paper. It suffices to say that the PDSQ statistic provides an estimate of S/N_0 through the use of the curves shown in Figure 2. These curves are plots of the mean value of the PDSQ statistic as a function of S/N_0 for the same three channel models as for Figure 1.

The *symbol error rate* (SER) is the relative frequency of errors among the symbols at the input to the decoder. As such, it is a measure of the quality of the received signal before any decoding is applied. The SER statistic that we consider is the number of differences between the symbols that are obtained by encoding the output of the decoder and the corresponding symbols at the decoder input. This SER statistic is a reliable estimate of the probability of symbol error on the channel provided there are not many errors in the decoder output. The mean values of the SER statistic for the same three channel models are shown as a function of S/N_0 in Figure 3.

VI. Estimate of S/N_0 from the Receiver Statistics

In this section we focus on the channel that has no multipath. Similar observations can be made for each of the other channel models. From Figure 1 we see that it is possible to obtain a reliable estimate of S/N_0 from the AGC statistic if $S/N_0 > 75$ dB-Hz. Below this point the curve flattens, and there is insufficient sensitivity in the mean of the AGC statistic to differences in S/N_0. From Figure 2 we see that it is possible to obtain a good estimate of S/N_0 from the PDSQ statistic if S/N_0 is between 60 and 80 dB-Hz. Outside this range, the mean of the PDSQ statistic is not as sensitive to changes in S/N_0, and so the estimation error can be large even if there is little error in the determination of the PDSQ statistic.

From Figure 3, we see that the SER statistic can be used to determine whether S/N_0 is below about 64 dB-Hz, because in this range the SER statistic is relatively large. In addition, the SER statistic can be employed to estimate S/N_0 in the range from about 64 to 70 dB-Hz. One use of the SER statistic is to replace the AGC statistic in the region where the AGC statistic is of little value (below about 70 to 75 dB-Hz). It turns out the the desired operating point for one application is $S/N_0 = 69$ dB-Hz, and the SER statistic provides a good way to verify that the system is operating near this point.

VII. Conclusions

An adaptive transmission system for direct-sequence spread-spectrum packet communication networks can select optimum transmission parameters based on statistics that are derived in the receiver and sent to the transmitter in acknowledgment packets or as part of a RTS/CTS exchange in the channel access protocol. The power, code rate, and symbol rate are adapted to match the path loss and multipath profile of the channel. Results are given for different multipath profiles to illustrate how the receiver statistics can be used to estimate the parameters needed for the adaptive transmission algorithm. The resulting adaptive transmission is designed to provide energy-efficient communications over multipath channels.

References

[1] M. B. Pursley and C. S. Wilkins, "Adaptive-rate coding for frequency-hop communications over Rayleigh fading channels," *IEEE Journal on Selected Areas in Communications*, vol. 17, no. 7, pp. 1224-1232, July 1999.

[2] M. B. Pursley and C. S. Wilkins, "Adaptation of the code rate and transmitter power in frequency-hop communications," submitted to the *IEEE Journal on Selected Areas in Communications*.

[3] J. H. Gass, M. B. Pursley, H. B. Russell, R. J. Saulitis, C. S. Wilkins, and J. S. Wysocarski, "Adaptive transmission protocols for frequency-hop radio networks," *Proceedings of the 1998 IEEE Military Communications Conference*, (Boston, MA), vol. 1, pp. 282-286, October 1998.

[4] M. B. Pursley, H. B. Russell, and J. S. Wysocarski, "Energy-efficient routing in frequency-hop networks with adaptive transmission," *Proceedings of the 1999 IEEE Military Communications Conference*, (Atlantic City, NJ), November 1999, to appear.

[5] S. Lin and D.J. Costello, Jr., *Error Control Coding: Fundamentals and Applications*. Englewood Cliffs, NJ: Prentice-Hall, 1983.

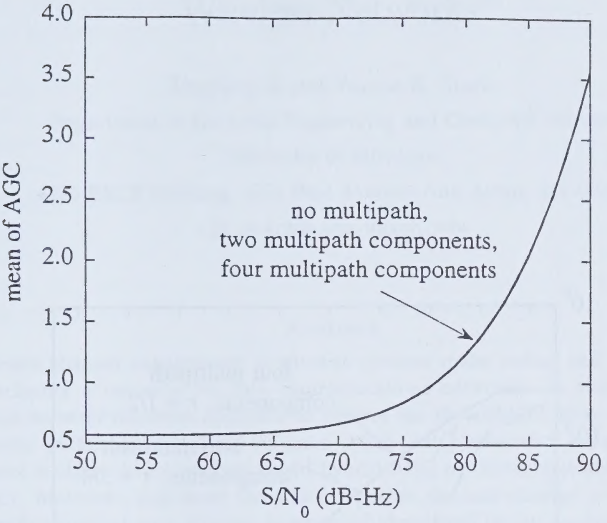


Fig. 1. The mean of the AGC statistic for all three channel models.

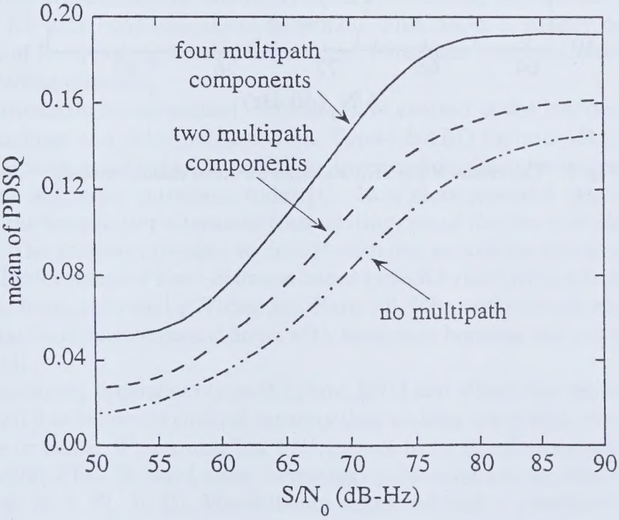


Fig. 2. The means of the PDSQ statistics for three channel models.

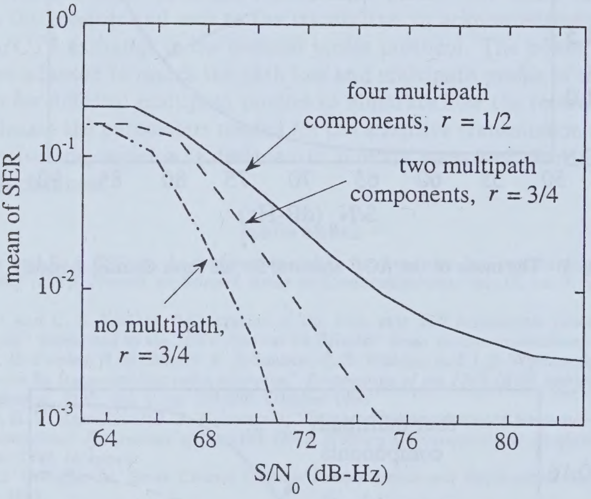


Fig. 3. The means of the SER statistics for three channel models.

Adaptive Redundancy Retransmission Protocols for Wireless Networks*

Tingfang Ji and Wayne E. Stark
Department of Electrical Engineering and Computer Science
University of Michigan
4315 EECS Building, 1301 Beal Avenue, Ann Arbor, MI 48109
`tji,stark@eecs.umich.edu`

Abstract

Severe channel impairments in wireless systems make coding and retransmission schemes a necessity for data communications networks. In this paper, we propose an error recursion approach to analyze the throughput, delay, and energy efficiency performance of Reed-Solomon coded ARQ schemes. A rate-adaptive protocol is shown to outperform type-I hybrid ARQ in throughput and energy efficiency. Moreover, significant the effect of packet size and channel correlation on the performance of both schemes is accurately predicted by our analysis.

1 Introduction

Severe channel impairments in wireless systems make coding and retransmission schemes a necessity for data communications networks. This paper is mainly concerned about the analysis of Reed-Solomon coded incremental redundancy retransmission schemes over correlated fading channels.

ARQ (automatic repeat request) schemes have evolved in the last decades to achieve better throughput and delay performances. Type-I HARQ (hybrid ARQ) provides more protection against poor channel conditions by encoding the information with both error detecting and error correcting codes [1]. In a more powerful type-II hybrid ARQ scheme [2], the transmitter alternates transmitting one of the two convolutional encoded codewords. The receiver attempts to decode each one separately and then the combined codeword. Performance of Reed-Solomon based type-II hybrid ARQ schemes over AWGN channels has been analyzed by Wicker and Bartz [3]. The performance of a similar system over a partial-band interference channel with frequency hopping was studied by Daraiseh and Baum [4].

Code combining generalizes type-II hybrid ARQ and allows the effective code rate to decrease until it is below the channel capacity thus making the system robust under severe interference or fading. Puncturing has been proven to be an effective technique to obtain rate compatible FEC (forward error correcting) codes that can be used in code combining protocols [5, 6, 7]. In [5], Mandelbaum suggested that a punctured Reed-Solomon

---

*This work was supported in part by the DARPA GloMo Project AO No. F383, AFRL contract number F30602-97-C-0314

codeword could be transmitted as the primary codeword and the remaining parity bits be used in retransmissions as incremental redundancy block, but no performance analysis has ever been published for this technique.

There has been extensive analysis of the performance of Reed-solomon codes on fading channels [8, 9, 10]. In [8], Bisch and Lutz studied the packet error rate in the non-interleaved Rayleigh channel. Performance analysis over finite state channels was studied in [9, 10].

In this paper, we use an error recursion technique to analyze the throughput, delay, and energy efficiency performance of punctured Reed-Solomon coded ARQ schemes. The challenge of the work is the complexity of computing packet error probability of *rate adaptive* Reed-Solomon code over correlated shadowing channel. The accuracy of this model also allows us to quantitively study the effect of block size and channel correlation on system performance.

The rest of the paper is organized as follows. In section II, the system and finite state channel model are introduced. In section III, an error recursion for type-I hybrid ARQ and rate-adaptive schemes correlated fading channel is described. In section IV, numerical results are presented and conclusions are given in section V.

2 System and Models

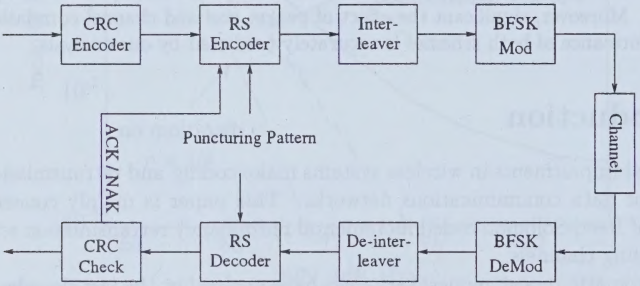


Figure 1: Block diagram of a rate-adaptive RS coded ARQ system

As shown in Figure 1, the transmitter is composed of a cyclic redundancy check (CRC) encoder, a Reed-Solomon encoder, a symbol interleaver, and a binary FSK modulator. The receiver consists of a square-law detector, a symbol deinterleaver, a Reed-Solomon decoder, and a CRC decoder.

The information bits are encoded by CRC encoder for error detection. The resulting bits are grouped into K Reed-Solomon symbols of size $\log_2(N) = m$ bits/symbol. Then the K symbols are encoded into a Reed-Solomon codeword of N symbols. The transmitter first transmits K information symbols, then transmits Δ symbols in each redundancy block upon request. Note that, here we assume the transmitter has infinite supply of source bits, and multiple codewords (or fractions of codewords if it is an incremental redundancy block) are sent through the channel within each packet. Under this assumption, we can further assume that the length of each packet is fixed (although the redundancy blocks for each block is not necessarily the same). Hence, even if Δ is small, the packet overhead can still be considered insignificant when compared with the total payload. At the receiver, if Reed-Solomon decoding failure occurs or CRC decoder does not check (undetected error probability of CRC is assumed to be negligible), a retransmission request is sent back to the transmitter via an error free feedback channel.

The error correcting capability of the code is increased at each retransmission. It is well known that for Reed-Solomon code, the increment of the error correcting capability due to a Δ symbol redundancy block is $\Delta/2$, For convenience of analysis, we assume Δ is even and $N-K$ divides Δ. When the jth transmission finishes, the total received symbols form a codeword of a $(K+(j-1)\Delta, K, (j-1)\Delta/2)$ punctured Reed-Solomon code. When the whole codeword of N symbols is transmitted, and there are still uncorrectable errors in the received word, the receiver discards previously received symbols and restarts the protocol.

When a noncoherent demodulator is employed at the receiver, the bit error probability P_b and Reed-Solomon code symbol error probability P_s are given by

$$P_b = \frac{1}{2}e^{-E/2N_0}, \quad P_s = 1-(1-P_b)^m. \tag{1}$$

2.1 Shadowing Channel Model

In the second channel model, we assume the transmitter receiver pair suffers from fast fading and lognormal shadowing, i.e., the logarithm of the received signal strength has a Gaussian distribution. The mean of the shadowing process is the path loss, which is assumed to be constant in our study. Since the shadowing is usually a very slow process compared to the data rate, the shadowing is assumed to be constant over a packet duration. Thus, the effect of shadowing on reception is analogous to varying the signal-to-interference ratio $\overline{\gamma}$ for each packet. The correlation between the shadowing factors of adjacent packets is modeled as an exponentially decreasing autocorrelation function [11]. Suppose the mobile velocity is v m/s, the correlation $R(T)$ of two packets which are T seconds apart is given by

$$R(T) = \sigma_s^2 \epsilon_D^{vT/D}$$

where σ_s is the standard deviation of the shadowing level, and $\sigma_s^2\epsilon_D$ is the correlation between two points separated by distance D. Under the fixed packet length assumption mentioned before, we can assume the SNR correlation between successive transmissions are constant no matter what the size of the redundancy block is. According to the 1700 MHz measurement in an urban area [11], σ_s was estimated to be 4.3 dB, and we will use this number throughout our analysis.

Fast fading is modeled as flat Rayleigh fading and independent from symbol to symbol. This model is accurate under the assumption of narrow band signaling and sufficiently large interleaver size. The bit error probability P_b of square-law detection of binary BFSK signal over Rayleigh fading channel is given by

$$P_b = \frac{1}{2+E/N_0}, \quad P_s = \sum_{l=1}^{m}\binom{m}{l}\frac{(-1)^{l+1}2^{-l+1}}{2+\frac{lE}{N_0}} \tag{2}$$

2.2 First-order Markov Model for Shadowing Channel

The time varying nature of the correlated channel can be captured by a first-order Markov model. If we quantize the SNR (signal-to-noise ratio) into L levels $\{U_1, U_2, \cdots, U_L\}$, a L state Markov model can be defined by the stationary distribution and the state-transition matrix. The steady state probability of state n can be obtained by integrating the density $p(x)$ over the region U_n,

$$P(U_n) = \int_{U_n} p(x)dx \tag{3}$$

where $p(x)$ is a Gaussian density function with mean zero and variance σ_s^2.

Consider two random variables corresponding to the values of the logarithm of the shadowing process at time t and $t-T$, say, Y and Z. The joint pdf of correlated Gaussian random variables is

$$p_{Y,Z}(y,z) = \frac{1}{2\pi\sigma_s^2\sqrt{1-\rho^2}} \exp\{-\frac{y^2 - 2\rho yz + z^2}{2(1-\rho^2)\sigma_s^2}\} \tag{4}$$

where $\rho = R(T)/\sigma_s^2$. The state transition matrix can then be easily obtained using

$$P(U_m|U_n) = \frac{\int_{U_m}\int_{U_n} p_{Y,Z}(y,z)dzdy}{\int_{U_n} p(z)dz}. \tag{5}$$

3 Error Recursion

Before we can address the problem, we define some notation and review the throughput analysis of type-I HARQ protocol with Reed-Solomon coding. Let

X_j = number of symbol errors in the resulting word from the 1st to the jth transmissions within the same codeword

S_j = number of symbol errors in the jth redundancy block

$P_c(j)$ = Prob {previous $j-1$ transmissions fail and the jth transmission leads to a successful decoding}.

Next, we develop the error recursion to calculate the performance for the rate adaptive scheme.

In this paper, we characterize the delay performance by the average number of retransmissions D

$$D = \sum_{j=1}^{\infty}(j-1)P_c(j). \tag{6}$$

The normalized throughput η is defined as the number of information bits successfully transmitted per channel use. η is given by

$$\eta = \frac{K}{\sum_{j=1}^{\infty} P_c(j)S_j}. \tag{7}$$

Ignoring the power consumed in data processing, the energy efficiency is proportional to the reciprocal of the effective bit energy E_b Joule/bit. For a certain background noise power spectrum density N_0, it is reasonable to define the relative energy efficiency as

$$\lambda(\gamma) = CN_0/E_b = \eta/\gamma \tag{8}$$

where γ is the SNR and C is some constant. Inspection of (6), (7), and (8), we note the all performance metrics are functions of $P_c(j)$.

$$\begin{aligned} P_c(j) &= P(X_j \le t_j, X_{j-1} > t_{j-1}, \cdots, X_1 > t_1) \\ &= \sum_{i=t_{j-1}+1}^{t_j} P(X_j = i, X_{j-1} > t_{j-1}, \cdots, X_1 > t_1) \\ &= \sum_{i=t_{j-1}+1}^{t_j} \sum_{m=1}^{L} P(X_j = i, X_{j-1} > t_{j-1}, \cdots, X_1 > t_1, \gamma_j = U_m) \end{aligned} \tag{9}$$

where t_j is the error correcting capability of the jth transmission, $0 \le t_j \le (N-K)/2$, γ_j is the SNR for the jth transmission. Let $P(j,i,U_m) = P(X_j = i, X_{j-1} > t_{j-1}, \cdots, X_1 > t_1, \gamma_j = U_m)$, the task of analyzing the system performance boils down to calculating $P(j,i,U_m)$. In the rest of this section, the recursive algorithms for calculating $P(j,i,U_m)$ are presented for HARQ and rate adaptive schemes over both AWGN and correlated fading channel.

For type-I hybrid ARQ over AWGN channel, $P_c(j)$ can be expressed in closed form

$$P_c(j) = \sum_{i=0}^{t} \binom{N}{i} P_s^i (1-P_s)^{N-i} (1 - \sum_{k=0}^{t} \binom{N}{k} P_s^k (1-P_s)^{N-k})^j \tag{10}$$

where $t = (N-K)/2$.

For rate adaptive scheme over AWGN channel,

$$P(j,i) = \sum_{k=0}^{\min(i-t_{j-1},\Delta)} q(k) P(j-1, i-k), \quad q(k) = \binom{\Delta}{k} P_s^k (1-P_s)^{\Delta-k} \tag{11}$$

The initial condition for the recursion is given by

$$P(1,i) = \binom{K}{i} P_s^i (1-P_s)^{K-i}$$

For type-I hybrid ARQ over correlated fading channel, let $P(j,U_m) = \sum_{i=1}^{t} P(j,i,U_m)$, then

$$P(j,U_m) = P(1,U_m)(1 - \sum_{k=1}^{j-1} \sum_{n} P(k,U_n) Cov_{|j-k|}(m,n)) \tag{12}$$

where

$$Cov_{|j-k|}(m,n) = \frac{P(\gamma_j = U_m, \gamma_k = U_n)}{P(\gamma_j = U_m) P(\gamma_k = U_n)}, \quad P(1,U_m) = \sum_{k=0}^{t} \binom{N}{k} P_s^k (1-P_s)^{(N-k)} P(\gamma_1 = U_m).$$

For rate adaptive system over correlated fading channel, let $V = (N-K)/\Delta + 1$ be the total number of redundancy blocks within one codeword. The recursion is given by

$$P(j,i,U_m) = \begin{cases} \sum_{k=0}^{\min(i-t_{j-1},\Delta)} q(k|U_m) \sum_l P(j-1, i-k, U_l) P(U_m|U_l) & j \bmod V \neq 1 \\ P(1,i,U_m)(1 - \sum_{s=1}^{j-1} \sum_{k=0}^{t_s} \sum_n P(s,k,U_n) Cov_{|j-s|}(m,n)) & j \bmod V = 1 \end{cases} \tag{13}$$

where

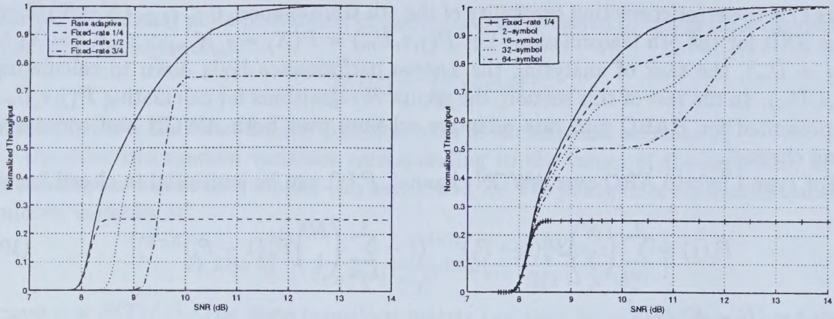
$$P(1,i,U_m) = \binom{K}{i} P_{s|m}^i (1-P_{s|m})^{K-i} P(\gamma_1 = U_m), \quad q(k) = \binom{\Delta}{k} P_{s|m}^k (1-P_{s|m})^{\Delta-k},$$

where $P_{s|m}$ is the symbol error probability conditioned on the fading leve U_m.

Since (10), (11), and (12) are special cases of (13), only a sketch of proof for (13) is provided in Appendix.

4 Numerical Results

In this section, we show the results obtained from the error recursion described in last section. A (256,64,96) Reed-Solomon code is considered.

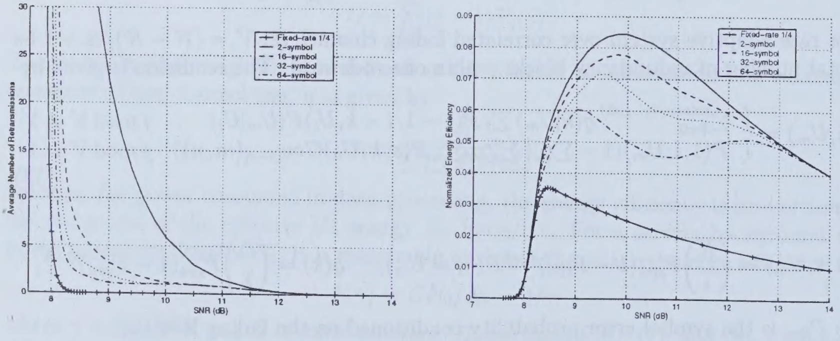


(a) Effect of code rate

(b) Effect of redundancy block size

Figure 2: Throughput comparison of type-I HARQ scheme with different code rate and rate-adaptive scheme with different redundancy block size over AWGN channel

In Figures 2 and 3, the performance of type-I hybrid ARQ and adaptive system over AWGN channel is compared. As can be observed in Figure 2(a), the adaptive scheme achieves better throughput than any fixed rate scheme. Additionally, in Figure 2(b), we show that the throughput performance of rate adaptive schemes with different redundancy block size is lower bounded by the performance of the rate 1/4 HARQ scheme. Moreover, we observe that only marginal gain can be obtained by reducing redundancy block size from 16 symbols to 2 symbols, while the large delay incurred by 2 symbol retransmission scheme is unacceptable as shown in Figure 3(a). In Figure 3(b), it is noted that the maximum energy efficiency achieved by rate adaptive schemes easily doubles that of rate-1/4 HARQ scheme.



(a) Average number of retransmissions

(b) Normalized energy efficiency

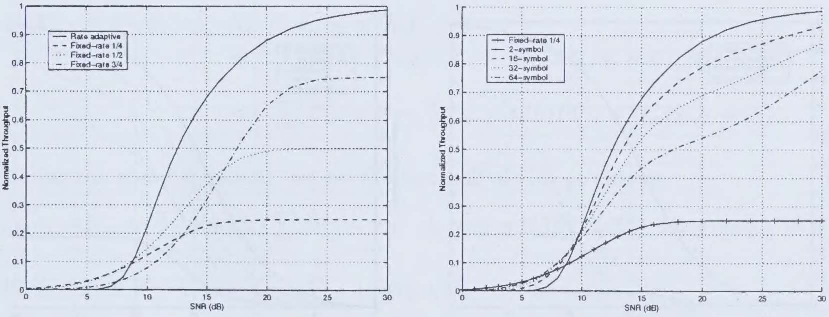
Figure 3: Delay, and energy efficiency of type-I HARQ scheme and rate-adaptive scheme over AWGN channel

In Figures 4 to 7, we plot the performance of two protocols over correlated shadowing channels with constant correlation coefficient between packets (0.5 for Figures 4 and 5, and 0.99 for Figures 6 and 7). Besides similar improvement as shown in AWGN channel, we note that the rate adaptive scheme does not perform as well as HARQ scheme when the adjacent packets are not very correlated and the SNR is very low (Figures 5(b) and 4(b)). This is due to the fact that for the fixed rate scheme, the whole codeword undergoes the same shadowing, which has standard deviation 4.3dB, while the averaging effect on the codeword in the rate adaptive scheme makes this standard deviation much smaller. Since at low average SNR, the chance of getting a successful decoding depends on the probability of getting a positive fading level, thus, the fixed rate protocol has better throughput than rate adaptive system.

Comparing Figures 4(b) and 6(b), it is noted that when the correlation coefficient between packets goes from 0.50 to 0.99, the HARQ system has almost 5dB loss in performance. For the rate adaptive scheme, larger correlation only makes the knee of the curve flatter.

5 Conclusion

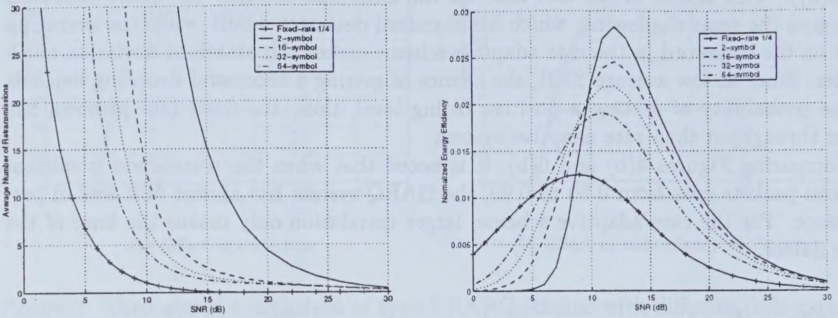
An error recursion is developed for performance analysis of rate adaptive ARQ schemes over AWGN and correlated fading channels. Compared with type-I hybrid ARQ protocol, the rate-adaptive protocol achieves much better performance under most channel conditions. Analysis shows that very fine redundancy blocks in rate-adaptive system introduce large delay while the return in throughput and energy efficiency is small. It is also noted that hybrid ARQ systems have large performance degradation in heavily correlated channel, while rate adaptive schemes are relatively insensitive to channel correlation. This analysis is also applicable to other channels including jamming, multiple access intereference, and fast fading.



(a) Effect of code rate

(b) Effect of redundancy block size

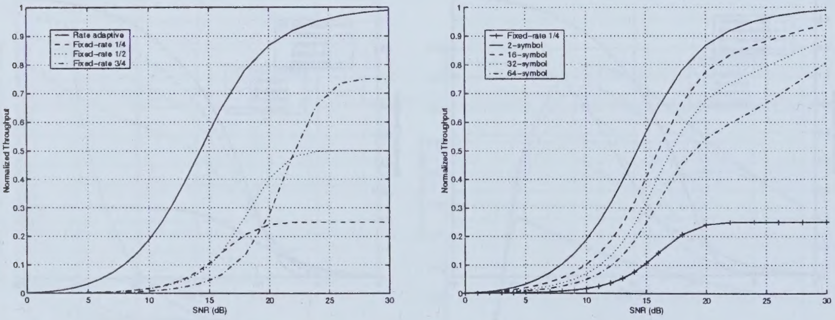
Figure 4: Throughput comparison of type-I HARQ scheme with different code rate and rate-adaptive scheme with different redundancy block size over correlated shadowing channel, $\rho = 0.50$



(a) Average number of retransmissions

(b) Normalized energy efficiency

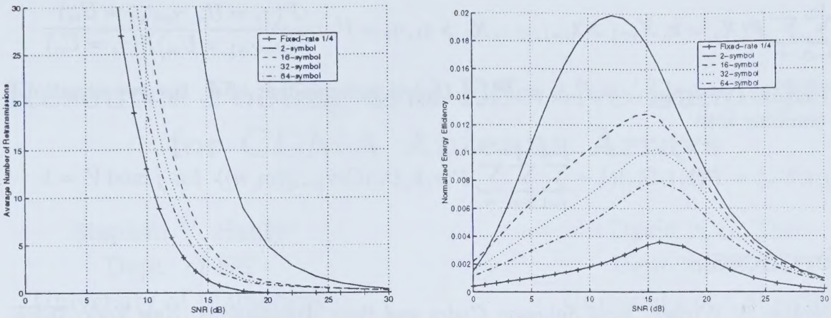
Figure 5: Delay, and energy efficiency of type-I HARQ scheme and rate-adaptive scheme over correlated shadowing channel, $\rho = 0.50$



(a) Effect of code rate

(b) Effect of redundancy block size

Figure 6: Throughput comparison of type-I HARQ scheme with different code rate and rate-adaptive scheme with different redundancy block size over correlated shadowing channel, $\rho = 0.99$



(a) Average number of retransmissions

(b) Normalized energy efficiency

Figure 7: Delay, and energy efficiency of type-I HARQ scheme and rate-adaptive scheme over correlated shadowing channel, $\rho = 0.99$

Appendix

In the appendix, we derive $P(j, i, U_m)$ for rate adaptive scheme over correlated fading channel (13).

For $j \bmod V \neq 1$, we have

$$\begin{aligned} P(j, i, \gamma_j = U_m) &= \sum_{k=0}^{\min(i-t_{j-1},\Delta)} P(S_j = k, X_{j-1} = i-k, \cdots, X_1 > t_1, \gamma_j = U_m) \\ &= \sum_k P(S_j = k|\gamma_j = U_m) P(X_{j-1} = i-k, \cdots, X_1 > t_1, \gamma_j = U_m) \end{aligned}$$

$$\begin{aligned} P(X_{j-1} = i-k, \cdots, \quad X_1 &> t_1, \gamma_j = U_m) \\ &= \sum_l P(X_{j-1} = i-k, \cdots, X_1 > t_1, \gamma_j = U_m, \gamma_{j-1} = U_l) \\ &= \sum_l P(j-1, i-k, \gamma_{j-1} = U_l) P(\gamma_j = U_m|\gamma_{j-1} = U_l). \end{aligned}$$

Combining the above equations, we get the first half of (13) as follows

$$P(j, i, \gamma_j = U_m) = \sum_k q(k|U_m) \sum_l P(j-1, i-k, \gamma_{j-1} = U_l) P(U_m|U_l) \quad \text{for } j \bmod V \neq 1.$$

When $j \bmod V = 1$, say, $j = wV + 1$, $P(j, i, U_m)$ can be written as

$$\begin{aligned} &P(wV+1, i, \gamma_{wV+1} = U_m) \\ &= P(X_{wV+1} = i, X_{wV} > t_{wV}, \cdots, X_1 > t_1, \gamma_{wV+1} = U_m) \\ &= P(X_{wV+1} = i|U_m) P(\gamma_{wV+1} = U_m) P(X_{wV} > t_{wV}, \cdots, X_1 > t_1|\gamma_{wV+1} = U_m) \\ &= P(X_{wV+1} = i, \gamma_{wV+1} = U_m)(1 - \sum_{s=1}^{wV} P(X_s \leq t_s, X_{s-1} > t_{s-1}, \cdots, X_1 > t_1|\gamma_{wV+1} = U_m) \end{aligned}$$

$$\begin{aligned} &P(X_s \leq t_s, X_{s-1} > t_{s-1}, \cdots, X_1 > t_1|\gamma_{wV+1} = U_m) \\ &= \sum_{k=0}^{t_s} \sum_n P(X_s = k, X_{s-1} > t_{s-1}, \cdots, X_1 > t_1, \gamma_s = U_n|\gamma_{wV+1} = U_m) \end{aligned}$$

$$= \sum_{k=0}^{t_s}\sum_{n} P(X_s = k, X_{s-1} > t_{s-1}, \cdots, X_1 > t_1, \gamma_s = U_n)\frac{P(\gamma_s = U_n, \gamma_{wV+1} = U_m)}{P(\gamma_{wV+1} = U_m)P(\gamma_s = U_n)}$$

Since $P(X_{wV+1} = i, \gamma_{wV+1} = U_m) = P(1, i, U_m)$ is independent of w, the two equations above combine into

$$P(j, i, U_m) = P(1, i, U_m)(1 - \sum_{s=1}^{j-1}\sum_{k=0}^{t_s}\sum_{n} P(s, k, U_n)\mathrm{Cov}_{|j-s|}(m, n)) \quad \text{for } j \bmod V = 1$$

References

[1] Stephen B. Wicker, *Reed Solomon Codes and their Applications*, New York: IEEE Press, 1994.

[2] S. Lin and P. S. Yu, "Hybrid ARQ scheme with parity retransmission for error control of satellite channels," *IEEE Transactions on Communications*, vol. COM-30, no. 7, pp. 1701–1719, July 1982.

[3] Stephen B. Wicker and Michael J. Bartz, "Type-II hybrid ARQ protocols using punctured MDS codes," *IEEE Transactions on Communications*, vol. 42, no. 2/3/4, pp. 1431–1440, February/March/April 1994.

[4] A-G. A. Daraiseh and C. W. Baum, "Packet Combining in Frequency-Hop Spread-Spectrum Communication Systems," *IEEE Transactions on Communications*, vol. 46, no. 1, pp. 23–33, January 1998.

[5] Mandelbaum D. M., "An Adaptive-Feedback Coding Scheme Using Incremental Redundancy," *IEEE Transactions on Informaion Theory*, vol. 20, pp. 388–389, May 1974.

[6] Joachim Hagenauer, "Rate-compatible punctured convolutional codes (RCPC codes) and their applications," *IEEE Transactions on Communications*, vol. 36, no. 4, pp. 389–400, Apr. 1988.

[7] Douglas N. Rowitch and Laurence B. Milstein, "Rate compatible punctured turbo (RCPT) codes in a hybrid FEC/ARQ system," in *Proceedings of the 1997 IEEE Global Telecommunications Mini- Conference*, Phoenix, AZ, Nov. 1997, IEEE, vol. 4, pp. 55–59.

[8] H Bischl and E. Lutz, "Packet error rate in the non-interleaved Rayleigh channel," *IEEE Transactions on Communications*, vol. 43, no. 2/3/4, pp. 1375–1382, February/March/April 1995.

[9] C. Pimentel and Blake I. F., "Non-interleaved Reed-Solomon coding performance on finite state channels," in *IEEE International Conference on Communication*, 1997, vol. 3, pp. 1493–1497.

[10] Michael J. Chu and Wayne E. Stark, "Effect of mobile velocity on communications in fading channels," *IEEE Transactions on Communications*, vol. 47, 1999, to appear.

[11] M. Gudmundson, "Correlation model for shadow fading in mobile radio systems," *Electronics Letters*, vol. 39, pp. 187–189, Aug. 1991.

Resource Pooling and Effective Bandwidths for CDMA Antenna Arrays

Stephen V. Hanly*
Dept. of EE
University of Melbourne
Australia
s.hanly@ee.mu.oz.au

David N.C. Tse†
Dept. of EECS
University of California
Berkeley, CA 94720, USA
dtse@eecs.berkeley.edu

Abstract

Prior work on spread spectrum systems with multi-user receivers obtained a characterization of the "effective interference" of one user on another, under the MMSE receiver, and an effective bandwidth characterization of the user capacity of a system employing the MMSE receiver [1]. In the present paper, we extend this result to the case of a spread spectrum antenna array system, in which each antenna experiences independent flat fading. We focus on the case in which the fading distributions to the antennas are identical, and show that a curious phenomenon of "resource pooling" arises: the multi-antenna system behaves like a system with only one antenna but with the processing gain the product of the processing gain of the original system and the number of antennas, and the received power of each user the sum of the received powers at the individual antennas.

1 Introduction

In recent years, there have been intense efforts in developing sophisticated multiuser techniques for wireless communications. A significant thrust of work has been on developing *multiuser* receiver structures which mitigate the interference between users in direct-sequence CDMA (DS-CDMA) systems. (See [2] for a comprehensive account of the state-of-the-art.) Unlike the conventional matched filter receiver used in the IS-95 CDMA system, these techniques take into account the structure of the interference from other users when decoding a user. Another important line of work is the development of signal processing techniques in systems with antenna arrays ([3], [4]).

Much work has already been undertaken on characterizing the performance of multiuser receivers, using measures such as asymptotic efficiency and near-far resistance [2]. A different point of view can be taken from a networking perspective. Rather than focusing

---

*S.V. Hanly gratefully acknowledges funding from an Australian Research Council small grant.

†The research of D.N.C. Tse is partially supported by a National Science Foundation Early Faculty Career Award.

on the performance of individual users, we ask the following question: given desired levels of performance (quality-of-service, or QoS) for each of the users in the network, what is the number of users that can be accommodated? A line of work towards a better understanding of this issue has recently been initiated in [1]. A network capacity analysis of linear multiuser receivers is done in the context of synchronous CDMA systems with *random* spreading sequences. The results are for large networks, asymptotic as both the number of users and processing gain grow. The QoS measure for each user is taken as the signal-to-interference ratio (SIR) achieved at the output of the multiuser receiver. A concept of *effective bandwidth* has emerged from this work as a succinct measure of network capacity: given a set of SIR requirements for the users in an uplink power-controlled cell, they can all be met if and only if the sum of the effective bandwidths of the users is less than a certain invariant quantity, which depends only on the total degrees of freedom. The effective bandwidth concept is based on a more general notion of *effective interference*, which captures the effect of an interferer, received at arbitrary power, on the user to be demodulated. While the concept of effective bandwidth holds for a single power controlled cell, the notion of effective interference can quantify the inter-cell interference effects as well.

In the present paper, we extend the above concepts to DS-CDMA systems with spatial diversity and the MMSE receiver. The spatial diversity we consider is that provided by the multiple antenna elements at a single base-station, often referred to as "microdiversity". We demonstrate that a curious phenomenon of "resource pooling" arises: the multi-antenna system behaves like a system with only one antenna but with the processing gain the product of the processing gain of the original system and the number of antennas, and the received power of each user the sum of the received powers at the individual antennas.

The effective bandwidth results described above hold in a *large* system with *random* spreading sequences. A natural question is whether an effective bandwidth characterization exists in a *finite-sized* system with *arbitrary* spreading sequences. We address this issue in Section 5.

2 Basic Multi-Antenna CDMA Model

In a DS-CDMA system, each of the user's information or coded symbols is spread onto a much larger bandwidth via modulation by its own *signature* or *spreading sequence*. We consider a sampled discrete-time baseband model for a symbol-synchronous multi-access CDMA system with K users, L receive antennas and processing gain N. The received signal at the lth antenna is given by

$$\mathbf{Y}(l) = \sum_{k=1}^{K} X_k \sqrt{T_k}\gamma_k(l)\mathbf{s}_k + \mathbf{W}(l), \tag{1}$$

where X_k is the symbol transmitted by user k at transmit power T_k, $\gamma_k(l)$ is the complex fading channel gain from user k to antenna l, $\mathbf{s}_k \in \mathbf{C}^N$ is the signature sequence of user k, $\mathbf{Y}(l) \in \mathbf{C}^N$, and $\mathbf{W}(l)$ is additive white Gaussian noise with variance σ^2, independent across l. The symbol energy $\mathbf{E}[|X_i|^2]$ is normalized to be 1. Here, we are assuming a flat fading channel model. Moreover, the channel gains are assumed to be circular symmetric, as is typical for a baseband model.

Let $\mathbf{X} := (X_1, \ldots, X_K)^t$, $\mathbf{Y} := (\mathbf{Y}(1)^t, , \ldots, \mathbf{Y}(L)^t)$, $\mathbf{W} := (\mathbf{W}(1)^t, \ldots, \mathbf{W}(L)^t)^t$, and $\mathcal{D} := \mathrm{diag}(T_1, \ldots, T_K), \mathcal{S}(l) := [\gamma_1(l)\mathbf{s}_1, \ldots \gamma_K(l)\mathbf{s}_K]$, $\mathcal{S} := [\mathcal{S}(1)^t, \ldots, \mathcal{S}(L)^t]^t$, i.e. the $\mathcal{S}(l)$s stacked one above the other. Also, let $\bar{\mathbf{s}}_k$ be the kth column of $\mathcal{S}$. Then we can write the overall channel as:

$$\mathbf{Y} = \sum_{k=1}^{K} X_k \sqrt{T_k} \bar{\mathbf{s}}_k + \mathbf{W}. \tag{2}$$

In vector form, the channel can be written as:

$$\mathbf{Y} = \mathcal{S}\mathcal{D}^{1/2}\mathbf{X} + \mathbf{W}$$

In the present paper, we focus on the case where the receive antennas are assumed to be placed at the same base-station. The diversity captured in this scenario is due to small-scale multipath fading effects. A reasonable model is to assume that the random channel gains $\gamma_k(l)$s are independent for all users k and antennas l, and for any given user, the gains to all the antennas are identically distributed. The crucial assumption here is the symmetry of the channel fading statistics with respect to the antennas. Such diversity is often called "microdiversity", as opposed to the more general "macrodiversity" model, in which the marginal distributions are not assumed identical. In the present paper, they *are* identical for a particular user, but the fading levels are not necessarily identically distributed across users. For example, some users may be close to the antenna array, and others far. We can think of the *distribution* of the $(\gamma_k(l))_{l=1}^L$ as being a function of the geographic location of user k with respect to the antenna array, on a *coarse* space scale, and the actual realizations of these random variables as a function of the user's random position as measured on a small space-scale.

We will also allow the transmit powers T_ks to depend on the magnitudes of the channel gains $\gamma_j(l)$ for all j and l, but independent of everything else. This models the use of power control.

Now let us consider the issue of multi-user receivers. A *linear* receiver $\mathbf{c}_1$ for user 1 generates a soft decision $\hat{X}_1 = \mathbf{c}_1^H \mathbf{Y}$ for X_1 based on the entire observation $\mathbf{Y}$. The key performance measure for a linear receiver is the output *signal-to-interference* ratio (SIR), defined by:

$$\mathrm{SIR}_1 = \frac{T_1 |\mathbf{c}_1^H \bar{\mathbf{s}}_1|^2}{\sigma^2(\mathbf{c}_1^H \mathbf{c}_1) + \sum_{k=2}^{K} T_k |\mathbf{c}_1^H \bar{\mathbf{s}}_k|^2}. \tag{3}$$

Consider now coherent receivers that demodulate $\mathbf{X}$ from $\mathbf{Y}$ with perfect knowledge of the signature sequences as well as the channel gains and transmit powers of the users. Among these receivers, the MMSE receiver minimizes the mean-square error as well as maximizes the SIR for all the users, given the signature sequences, channel gains and transmit powers.

The MMSE receiver for user 1 is given by:

$$\begin{aligned}
\mathbf{c}_1 &= (\mathrm{E}[\mathbf{Y}\mathbf{Y}^H])^{-1}\mathrm{E}[X_1^* \mathbf{Y}] && (4)\\
&= \sqrt{T_1}(\mathcal{S}\mathcal{D}\mathcal{S}^H + \sigma^2 I)^{-1}\bar{\mathbf{s}}_1 && (5)\\
&= \text{constant} \cdot (\mathcal{S}_1\mathcal{D}_1\mathcal{S}_1^H + \sigma^2 I)^{-1}\bar{\mathbf{s}}_1 && (6)
\end{aligned}$$

where $\mathcal{S}_1$ is the matrix obtained by removing the column $\bar{s}_1$ from $\mathcal{S}$, and $\mathcal{D}_1 = \text{diag}(T_2, \ldots, T_K)$ The last step follows from the use of the matrix inversion lemma. The expectation is taken by averaging all quantities that are unknown to the receiver, in this case the unknown symbols of the users, and the background white noise.

The SIR of the MMSE receiver for user 1 is given by the expression

$$\text{SIR}_1 = T_1 \bar{s}_1^H (\mathcal{S}_1 \mathcal{D}_1 \mathcal{S}_1^H + \sigma^2 I)^{-1} \bar{s}_1, \tag{7}$$

Observe that the SIRs achieved by each user are functions of the signature sequences, channel gains and transmit powers of *all* users. There is in fact a functional relationship between the SIR attained by a user and its MMSE ([5]):

$$SIR_1 = 1 - \frac{1}{\text{MMSE}_1} \tag{8}$$

While (7) can be numerically computed given specific signature sequences, channel gains and transmit powers, the qualitative dependence of performance on system parameters such as the number of users, the processing gain, the number of antennas, or the received power profile is not clear. To obtain more insight, we will assume a *random* signature sequence model: the chip values of the sequences are i.i.d. circular symmetric complex Gaussian random variables with mean zero and variance $1/N$, and the sequences of different users are chosen independently [1]. The SIR depends on the realization of the random sequences as well as the channel gains, and is therefore also a random variable. It will be shown, however, that in the cases of interest, this random variable converges to a *deterministic* quantity in a large system, and thus provides a sequence-independent performance measure of the system. It should be emphasized that although the sequences are chosen randomly, knowledge of the sequences is assumed at the MMSE receiver so that interference suppression can be performed.

3 Resource Pooling and Effective Interference

The following is the main result in this section, yielding the asymptotic performance of the MMSE receiver in a system with large processing gain and many users, but fixed number of antennas.

Theorem 1 *Let $P_k = T_k \sum_{l=1}^{L} |\gamma_k(l)|^2$ be the sum of the received powers of user k. Assume that almost surely the empirical distribution of $(P_1, \ldots, P_K)$ converges weakly to a limiting distribution F as N goes large, and that the P_ks are uniformly bounded for all k and N* [2]. *Then if $N, K \to \infty$ with $K/N \to \alpha$ but L fixed, SIR_1/P_1 converges in probability to a deterministic constant a, where a is the unique positive solution to the fixed-point equation:*

$$a = \frac{1}{\sigma^2 + \frac{\alpha}{L} E\left[\frac{P}{1+Pa}\right]} \tag{9}$$

and P is a random variable having distribution F.

---

[1] We conjecture that all our results hold for general i.i.d. chip distribution.

[2] This latter assumption is a technicality to simplify the proofs, but we believe that it is not really necessary.

Proof See [6]. ◦

This result says that in a wideband system with many users, the SIR of a user does not depend on the specific realization of the signature sequences, the channel gains and the transmit powers. The SIR is a function of the user's own received powers at the antennas and depends on the the interferers' received powers only through the limiting empirical distribution of the P_ks. In a sense, there is an averaging of the effects across the large number of interferers. The convergence of the empirical distribution of the received power is a statistical regularity assumption and is necessary for such averaging to occur. It is satisfied, for example, when the transmit power T_k of any user k depends on the channel gains for that user only. In this case, the total received power of any user has the same distribution which coincides with the limiting distribution F.

When there is only one antenna element, P_k is simply the received power of user k. Theorem 1 therefore has the nice interpretation that for any fixed number of antennas L, the limiting performance of the MMSE receiver is the same as that for a system with a single antenna, having processing gain LN and with the received power of each user the sum of the received powers at the individual antennas. This is a form of *resource pooling*: all the degrees of freedom of the individual antennas are pooled together into a single resource and the system behaves like a single antenna system.

One can consider the multi-antenna spread-spectrum system as one with LN degrees of freedom, given by N units of processing gain per antenna. The "super" signature sequence of user k, of length LN, is then the signature sequence $\bar{\mathbf{s}}_k$, with the same signature sequence repeated L times and multiplied by the path gains $\gamma_k(l)$s. One might imagine that this repetition would lead to some loss of degrees of freedom; the resource pooling interpretation of Theorem 1 however shows that there is actually no loss.

Due to the resource pooling phenomenon, we can treat the system with microdiversity as one with a single antenna element, to which we can immediately apply the interpretation of effective interference derived in [1] : in a large system, the SIR of user 1 approximately satisfies the fixed point equation:

$$\mathrm{SIR}_1 = \frac{P_1}{\sigma^2 + \frac{1}{LN}\sum_{k=2}^{K} I(P_k, P_1, \mathrm{SIR}_1)} \tag{10}$$

where $I(P_k, P_1, \beta) := P_1 P_k/(P_1 + P_k\beta)$. Note that to determine the asymptotic performance we need to solve the fixed-point equation, which can be done numerically. However, to determine whether a desired SIR for user 1, β_1, is achievable, we can substitute β_1 into the right hand side of (10), and check if

$$\frac{P_1}{\sigma^2 + \frac{1}{LN}\sum_{k=2}^{K} I(P_k, P_1, \beta_1)} \geq \beta_1.$$

(This is due to the fact that the right hand side of (10) is a monotonic function of SIR_1. See [1] for details) In doing so, we provide a way to decouple the interference effects of each user, and we can define the *effective interference* of an interferer of received power P_k on user 1, as $I(P_k, P_1, \mathrm{SIR}_1)$.

4 Resource Pooling and Effective Bandwidths

Now let us consider the case in which users have particular SIR *targets*, *i.e.* user k has a target of β_k for its SIR. To achieve such a SIR target the user must control its transmit power, and we wish to find a necessary and sufficient condition on the targets β_ks for this to be possible, asymptotically, in a large system. This yields the user capacity region of the system. Here, we allow the power control of a user to be possibly a function of the magnitudes of the channel gains of all users, but not of the signature sequences. However, we will see in Section 5 that in a certain sense the capacity does not increase even when the powers can depend on the signature sequences.

We make the following statistical assumptions on the targets β_ks:

1) Almost surely, the joint empirical distribution of

$$(\beta_1, \gamma_1(1), \ldots, \gamma_1(L)), \ldots, (\beta_K, \gamma_K(1), \ldots, \gamma_K(L))$$

converges to some limiting distribution H as K grows;

2) The limiting distribution H satisfies

$$H(\beta, \gamma(1), \ldots, \gamma(L)) = H_1(\beta) H_2(\gamma(1), \ldots, \gamma(L)).$$

The second assumption says that the SIR requirement of a user is independent of its channel gains. Note that a basic assumption in Section 2 also requires that the marginal distributions of the $\gamma(l)$s are identical, so we assume that here as well.

Using Theorem 1, it can be seen that to meet the target SIRs, asymptotically, the transmit power T_k of user k should satisfy:

$$P_k a = \beta_k \tag{11}$$

where a satisfies the fixed point equations (9), and P has the limiting empirical distribution of the received powers $(P_1, \ldots, P_K)$. This limiting distribution can in turn be calculated from (11) and assumptions 1 and 2. We note that Theorem 1 implies that a is strictly positive. But from (11), we have that

$$\frac{P_k}{1 + P_k a} = \frac{1}{a} \frac{\beta_k}{1 + \beta_k}$$

and hence

$$\mathbf{E}\left[\frac{P}{1 + Pa}\right] = \frac{1}{a}\mathbf{E}\left[\frac{\beta}{1 + \beta}\right],$$

where β has the limiting distribution, H_1, of the SIR targets of the users. Define

$$B_{\text{mmse}} := \mathbf{E}[\frac{\beta}{1+\beta}] = \lim_{K \to \infty} \frac{1}{K} \sum_{k=1}^{K} \frac{\beta_k}{1 + \beta_k}.$$

It follows from the above calculations that a necessary condition that the given SIR targets can be met asymptotically is that the fixed point equation:

$$a = \frac{1}{\sigma^2 + \frac{\alpha}{L} B_{\text{mmse}} \frac{1}{a}} \tag{12}$$

has a strictly positive solution for a. But (12) can be solved for a directly; the solution is given by:

$$a^* = \frac{1}{\sigma^2}(1 - \frac{\alpha}{L} B_{\text{mmse}}), \tag{13}$$

and the condition for strict positivity is simply that $\alpha B_{\text{mmse}} < L$. Conversely, suppose that a^* is defined by (13), and is strictly positive. Let us now choose the transmit power for user k to be:

$$T_k^* = \frac{\beta_k}{a^* \sum_{l=1}^{L} |\gamma_k(l)|^2}$$

With this choice of transmit powers and the fact that a^* satisfies (12), it follows that a^* also satisfies (9), with the expectation over the limiting empirical distribution of P induced by the choice of transmit powers $\{T_k^*\}$. It follows from Theorem 1 that, due to the uniqueness of the solution to (9), the limiting SIR of user k is precisely β_k.

We can summarize the above development in the following theorem.

Theorem 2 *Let $(\beta_k)_{k=1}^{\infty}$ be the SIR targets of the users, and suppose assumptions 1 and 2 hold. Define*

$$B_{\text{mmse}} := \lim_{k \to \infty} \frac{1}{K} \sum_{k=1}^{K} \frac{\beta_k}{1 + \beta_k}$$

Then

1. *If $\alpha B_{\text{mmse}} > L$ then there is no way to assign transmit powers $(T_k)_{k=1}^{\infty}$ such that the users asymptotically achieve SIR targets $\beta_k, k = 1, 2, \ldots$.*

2. *If $\alpha B_{\text{mmse}} < L$, then the users' target SIRs can be asymptotically met, with transmit powers given by:*

$$T_k = \frac{\beta_k}{a^*} (\sum_{l=1}^{L} |\gamma_k(l)|^2)^{-1} \tag{14}$$

where a^* is given by (13).

Thus, the condition $\alpha B_{\text{mmse}} < L$ characterizes the interference-limited user capacity of the system. The above theorem also provides us with a notion of *effective bandwidth.* There are a total of LN degrees of freedom provided by spreading and the antenna array in the limit as $N \to \infty$. If user k achieves its target SIR of β_k, then it occupies an effective bandwidth of $\frac{\beta_k}{1+\beta_k}$ degrees of freedom, and to achieve the SIR targets, the sum of effective bandwidths must be less than the total number of degrees of freedom.

5 A Deterministic Formulation

In the previous sections, we have focused on an asymptotic model where the signature sequences are randomly chosen and the number of users and the processing gain are both large. Moreover, the power control was allowed to be based on only the channel gains but not the specific signature sequences. In the present section, we consider a *finite-sized* system with *arbitrary* signature sequences, and look at the capacity under the MMSE receiver. We show that the asymptotic capacity results we derived in the preceding sections, actually hold in the finite case, provided we can control transmit powers as a function of the given signature sequences. Our results will also provide insight into why the asymptotic results hold for random sequences.

Given a set of signature sequences $\mathbf{s}_1, \ldots, \mathbf{s}_K$, channel gains $\gamma_k(l)$s, and transmit powers T_ks, the SIR for user 1 is given by eqn. (7), with the matrix $\mathcal{S}$ defined there. We assume that user k has a required SIR of β_k.The *power control problem* is to find transmit powers $T_1, \ldots, T_k$ for the users in such a way that each user attains its required SIR. The necessary and sufficient conditions for a solution to this problem characterize the capacity region.

The following theorem gives us the effective bandwidth characterization of the capacity region of the finite system under discussion. In what follows, let SIR(**T**) denote the vector of attained SIRs when the transmit power vector is $\mathbf{T} = (T_1, \ldots, T_K)^t$. For a subset of users $U \subseteq \{1, 2, \ldots, K\}$, let $\mathcal{S}(U)$ be the $LN \times |U|$ matrix consisting of the columns of $\mathcal{S}$ corresponding to users in U, and $\mathcal{D}(U)$ be the diagonal matrix of transmit powers of users in U.

Theorem 3 *The constraints*

$$\sum_{k \in U} \frac{\beta_k}{1+\beta_k} < \text{rank}\ (\mathcal{S}(U)) \quad \forall U \subseteq \{1, 2, \ldots, K\} \tag{15}$$

are necessary and sufficient for there to exist a positive transmit power vector $\mathbf{T}^*$ *that solves the SIR equations*

$$SIR_k(\mathbf{T}^*) = \beta_k \quad k = 1, 2, \ldots, K \tag{16}$$

If a solution exists then it is unique.

Proof See [6]. ○

The reason why we have to consider subsets of users, and hence have an exponential number of constraints in the number of users, is because signature sequences can be allocated *badly.* Theorem 3 shows that the number of degrees of freedom available to a subset of users is given by the span of their signature sequences. Therefore, even if users outside a given subset U are allocated zero power, users within U can be unable to achieve their targets if the sum of their effective bandwidths is larger than the total degrees of freedom available to them. Signature sequences can be said to have been allocated "badly" if a large number of users have "similar" signature sequences, *i.e.* ones that occupy a low-dimensional space. To characterize when signature sequences are "good", we have the following Corollary and Proposition.

Corollary 4 *Assume that* $K \geq LN$. *Suppose that the matrix* $\mathcal{S}$ *has the property:*

$$\textit{every subset of } LN \textit{ columns are linearly independent.} \tag{17}$$

Then a necessary and sufficient condition for there to be a positive transmit power vector $\mathbf{T}^*$ *that solves the SIR equations in (16) is that*

$$\sum_{i=1}^{K} \frac{\beta_i}{1+\beta_i} < LN \tag{18}$$

Under what conditions on the original signature sequences $\mathbf{s}_1, \ldots, \mathbf{s}_K$ does $\mathcal{S}$ have the property (17)? Let $\mathcal{T} = [\mathbf{s}_1, \ldots, \mathbf{s}_K]$ and for any subset of users U, $\mathcal{T}(U)$ be the matrix the columns of which are the signature sequences of users in U (in increasing order, say.)

Proposition 5 *Assume* $K \geq LN$. *If every set of users of size* LN *can be partitioned into* L *disjoint subsets* $U_1, \ldots, U_L$, *each of size* N, *such that* $\mathcal{T}(U_1), \ldots, \mathcal{T}(U_L)$ *are all full rank, then property (17) is satisfied almost surely in the (random) channel gains* $\gamma_k(l)$.

Proof See [6] ○

It is worth noting that if the elements of the signature sequences are drawn randomly and independently from the same *continuous* distribution then the assumed condition in Proposition 5 will hold with probability 1. Moreover, even when the elements are discrete, say taking on values +1 or −1, it can be shown that the condition will hold almost surely in a large system as $N, K \to \infty$, $K/N \to \alpha$. This helps to explain why only one constraint arises in the random sequences case (Theorem 2). It is indeed interesting to observe that the feasible condition (18) is very similar to the capacity constraint for random sequences in a large system, as in Theorem 2. However, using random sequences, the power control to achieve the same capacity region is much simpler than for arbitrary sequences: the transmit power of a user can be chosen as a function of only its SIR requirement and its own channel gains, and this is a consequence of deeper random matrix properties.

6 Conclusions

Our main results are the limiting SIR performance and the effective bandwidth characterization of capacity for a DS-CDMA system with random spreading, spatial diversity and MMSE multiuser receiver. The asymptotics are for large processing gain N and large number of users K, but fixed number of antennas L. We focussed on the case of microdiversity, in which the channel gains to the antenna elements are independent and identically distributed for each user, and observed a curious "resource pooling" effect: one can consider the microdiversity array as a single antenna with processing gain LN, and pool the received powers of a user by adding up its received power at each of the individual antenna element. Generalizing the results in [1], we provided an effective bandwidth characterization of the user capacity region for the MMSE receiver. The effective bandwidth for a user with SIR target β is given by: $e(\beta) = \dfrac{\beta}{1+\beta}$ with the capacity region asymptotically given by

$$\sum_{k=1}^{K} e(\beta) < LN.$$

Thus, in a system with processing gain N, and L antennas, the overall number of degrees of freedom is LN.

In Section 5, we considered the finite case, with *arbitrary* signature sequences, and again characterized the capacity region, in terms of effective bandwidths. These results require that users are able to control their transmit powers as a function of their signature sequences, and we note that this was not allowed in our asymptotic, random signature sequence model. Also, the dimensionality of the space spanned by the signature sequences is an important issue in the finite case. Nevertheless, similar effective bandwidth results arise, and provide insight into the structure of our asymptotic results for random sequences.

We have also extended our results to the case of macrodiversity in [6], where the performance of the decorrelator and matched filter receivers are also characterized. However, in the macrodiversity case, we have only proved results so far under the assumption that each antenna sees a different randomly selected signature sequence *from the same user*, rather than the more correct repeated signature sequence model of the present paper, so further work is required to give a complete generalization of the present results to the macrodiversity scenario.

References

[1] D. Tse and S. Hanly, "Linear multiuser receivers: Effective interference, effective bandwidth and capacity," *IEEE Trans. Inform. Theory*, vol. 45, pp. 641–657, Mar. 1999.

[2] S. Verdú, *Multiuser Detection.* Cambridge, UK: Cambridge University Press, 1998.

[3] J. Winters, J. Salz, and R. Gitlin, "The impact of antenna diversity on the capacity of wireless communication systems," *IEEE Trans. Commun.*, vol. 42, pp. 1740–1751, February/March/April 1994.

[4] A. Paulraj and B. Papadias, "Space-time processing for wireless communications," *IEEE Signal Processing Magazine*, vol. 14, pp. 49–83, Nov. 1997.

[5] U. Madhow and M. L. Honig, "MMSE interference suppression for direct-sequence spread-spectrum CDMA," *IEEE Trans. Commun.*, vol. 42, pp. 3178–3188, Dec. 1994.

[6] S. Hanly and D. Tse, "Resource pooling and effective bandwidths in cdma systems with multiuser receivers and spatial diversity," *Submitted to IEEE Trans. Inform. Theory*, 1999.

Routing for Maximum System Lifetime in Wireless Ad-hoc Networks *

J.-H. Chang and L. Tassiulas

Electrical and Computer Engineering Dept. & Institute for Systems Research
University of Maryland
College Park, MD 20742
{jhchang,leandros}@isr.umd.edu

Abstract

An ad-hoc network of wireless static nodes is considered as it arises in a rapidly deployed, sensor based, monitoring system. Information is generated in certain nodes and needs to reach some designated gateway node. Each node may adjust its power within a certain range that determines the set of possible one hop away neighbors. Traffic forwarding through multiple hops is employed when the intended destination is not within immediate reach. The nodes have limited initial amounts of energy that are consumed in different rates depending on the power level and the intended receiver. We propose algorithms to select the routes and the corresponding power levels such that the time until the batteries of the nodes drain-out is maximized. The algorithms are local and amenable to distributed implementation. When there is a single power level, the problem is reduced to a maximum flow problem with node capacities and the algorithm converges to the optimal solution. When there are multiple power levels then the achievable lifetime is within 90% of the optimal (that is computed by linear programming) most of the time. It turns out that in order to maximize the lifetime, the traffic should be routed such that the energy consumption is balanced among the nodes in proportion to their energy reserves, instead of routing to minimize the absolute consumed power.

1 Introduction

Consider a group of wireless static nodes randomly distributed in a region as in Figure 1, where each node has a limited battery energy supply used mainly for the transmission of data. Assume that at each node some type of information is generated, and the information needs to be delivered to some nodes designated as gateway nodes. These wireless nodes are assumed to have the capability of packet forwarding, i.e., relaying an incoming packet to one of its neighboring nodes, and the transmit power level can be adjusted to a level appropriate for the receiver to be able to receive the data correctly if the receiver is within the transmission range. Upon or before a new arrival of information either generated at the node itself or forwarded from the other nodes, routing decision has to be made so that the node knows which of its neighboring nodes to send its data to.

---

*Prepared through collaborative participation in the Advanced Telecommunications/Information Distribution Research Program (ATIRP) Consortium sponsored by the U.S. Army Research Laboratory under the Federated Laboratory Program, Cooperative Agreement DAAL01-96-2-0002.

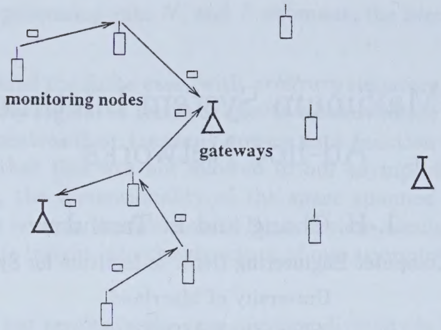


Figure 1: A multi-hop wireless ad-hoc network is depicted where the information generated at the monitoring nodes are delivered to the gateway nodes.

Note that the routing decision and the transmission energy level selection are intrinsically connected in this power-controlled ad-hoc network since the power level will be adjusted depending on the location of the next hop node.

An example scenario for this type of wireless ad-hoc network may include a wireless sensor network where the sensors gather acoustic, magnetic, or seismic information and send the information to its gateway node which has more processing power for further processing of the information or has larger transmission range for the delivery of the information to a possibly larger network for retrieval by a remote user.

Most of the previous works on routing in wireless ad-hoc networks deal with the problem of finding and maintaining correct routes to the destination during mobility and changing topology [1, 4, 9]. In [1, 4], the authors presented a simply implementable algorithm which guarantees strong connectivity and assumes limited node range. Shortest-path algorithm is used in this strongly connected backbone network. However, the route may not be the minimum energy solution due to possible omission of the optimal links at the time of the backbone connection network calculation. In [9], the authors developed a dynamic routing algorithm for establishing and maintaining connection-oriented sessions which uses the idea of predictive re-routing to cope with the unpredictable topology changes. Some other routing algorithms in mobile wireless networks can be found in [13, 10, 7, 12], which, as the majority of routing protocols in mobile ad-hoc networks do, use shortest-path routing where the number of hops is the path length.

The problem of minimum energy routing has been addressed before in [1, 4, 14, 8, 6, 16, 15], and [5]. The approach in those works was to minimize the consumed energy to reach the destination. If all the traffic is routed though through the minimum energy path to the destination the nodes in that path will be drain-out of batteries quickly while other nodes, which perhaps will be more power hungry if they forward traffic through them, will remain intact. Instead of trying to minimize the consumed energy, we address directly the performance objective of interest, that is to maximize the lifetime of the system. In order to achieve this, it turns out that it is more important to route the traffic such that the energy consumption is balanced among the nodes in proportion to their available energy instead of minimizing the absolute consumed power. In our study the topology of the network is static and the routing accounts to finding the traffic splits that balance optimally the energy consumption. Hence the results are applicable to networks which are either static, like the sensor networks we mentioned earlier, or their topology changes slowly enough such that there is enough time for optimally balancing the traffic in the periods between successive topology changes.

This paper is organized as follows: In section 2, the problem is formulated. In section 3, we propose a flow redirection algorithm. In section 4, we propose the maximum residual energy path routing algorithm. In section 5, random graphs are generated in order to evaluate the performances of these three algorithms.

2 Flows, Optimal Energy Consumption and Transmission Power Levels

Consider a directed graph $G(N, A)$ where N is the set of all nodes and A is the set of all directed links (i, j) where $i, j \in N$. Let S_i be the set of nodes that can be reached by node i with a certain power level in its dynamic range. We assume that link (i, j) exists if $j \in S_i$. Let each node i have the initial battery energy E_i, and let Q_i be the rate at which information is generated at node i. The transmission energy required for node i to transmit a bit to its neighboring node j is e_{ij}, and the rate at which information is transmitted from node i to node j is called the flow q_{ij}. We have a set of origin nodes O where the information is generated, and a set of destination nodes D among which any node can be reached in order for the information transfer be considered done.

The conservation of flow condition at each node i is assumed, i.e., the sum of all incoming flow must be the same as the sum of all outgoing flow,

$$\sum_{j:\, i \in S_j} q_{ji} + Q_i = \sum_{k \in S_i} q_{ik}, \quad \forall\, i \in N - D. \tag{1}$$

Subject to this constraint, we would like to maximize the lifetime of the network. The time it takes for the battery of node i to drain out under flow $\mathbf{q} = \{q_{ij}\}$ is given by

$$T_i(\mathbf{q}) = \frac{E_i}{\sum_{j \in S_i} e_{ij} q_{ij}}. \tag{2}$$

System lifetime : The lifetime of the system under flow q is defined as the minimum battery lifetime over all nodes,

$$T_{sys}(\mathbf{q}) = \min_{i \in N} T_i(\mathbf{q}). \tag{3}$$

Using the definition of the system lifetime, we can express the problem as follows:

$$\max_{\mathbf{q}} T_{sys}(\mathbf{q}), \quad \text{or equivalently,} \quad \max_{\mathbf{q}} \min_{i \in N} \frac{E_i}{\sum_{j \in S_i} e_{ij} q_{ij}}, \tag{4}$$

subject to (1), the flow conservation.

One can observe that the problem of maximizing the system lifetime given the information generation rates is equivalent to a linear programming problem[11] given by

$$\begin{aligned}
& & \text{Maximize} \quad T & & & (5)\\
\text{s.t.} & & \hat{q}_{ij} \geq 0, & \quad \forall j \in S_i,\ \forall i \in N - D, & & (6)\\
& & \textstyle\sum_{j \in S_i} e_{ij} \hat{q}_{ij} \leq E_i, & \quad \forall i \in N - D, & & (7)\\
& & \textstyle\sum_{j:\, i \in S_j} \hat{q}_{ji} + T Q_i = \sum_{k \in S_i} \hat{q}_{ik}, & \quad \forall i \in N - D, & & (8)
\end{aligned}$$

where $\hat{q}_{ij} = T q_{ij}$ is the amount of information transfered during T from node i to j.

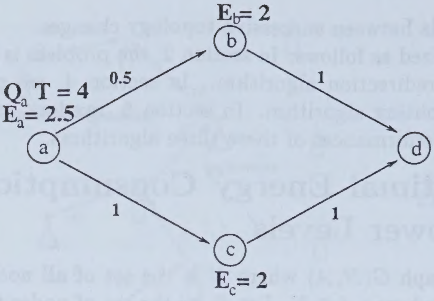


Figure 2: Counterexample to sufficiency of feasibility.

The formulation can be viewed as a variation of the conventional maximum flow problem with node capacities[3] in the special case where the transmitted power level at each node is fixed regardless of its next hop node, i.e., $e_{ij} = e_i$, for all $j \in S_i$, then (7) can be replaced by a set of node constraints

$$\sum_{j \in S_i} \hat{q}_{ij} \leq E_i / e_i, \ \forall i \in N - D. \tag{9}$$

In the following, we discuss conditions for the feasibility in order to attack the above problem from a different angle. For a set of nodes V, assume that each node $i \in V$ has the informatiot generation rate Q_i which needs to be delivered out of V. For a node $i \in V$ let e_i^V be the least energy expenditure for transporting an information unit out of V. If there is no outgoing link of i through which information can be forwarded out of V, $e_i^V = \infty$ by convention. The necessary feasibility condition for the system lifetime to be longer than T is given by

$$\sum_{i \in V} TQ_i \leq \sum_{i \in V} \frac{E_i}{e_i^V}. \tag{10}$$

Note that $\frac{E_i}{e_i^V}$ is the maximum amount of information that can flow out of V via node i.

The following counterexample shows that the necessary condition is not sufficient. In Figure 2 one can verify that the necessary feasibility condition is met. However, the flow is not feasible since the total energy constraint at node a corresponding to (7),

$$0.5\hat{q}_{ab} + (4 - \hat{q}_{ab}) \leq 2.5, \tag{11}$$

requires $\hat{q}_{ab} \geq 3$, which together with the flow conservation condition at node b violates the total energy constraint at node b. It can be verified that if the energy expenditure through all the outgoing links of a sensor were the same then the necessary condition would be sufficient as well. In other words, if the transmit power levels are fixed, then the condition becomes both necessary and sufficient for feasibility.

3 Flow Redirection Algorithm

In this section, we propose a flow redirection algorithm (FR), which is motivated by the following observation.

Theorem 1 (Necessary optimality condition) *If the minimum lifetime over all nodes is maximized then the minimum lifetime of each path flow from the origin to the destination with positive flow has the same value as the other paths.*

Proof: Assume that the minimum lifetime over all nodes is maximized. If we further assume that the minimum lifetimes of the paths with positive flow to the destination are not all identical then there is a path with positive flow whose minimum lifetime is smaller than that of any other path flows. We can always increase the minimum lifetime of this path, which is also the minimum lifetime over all nodes, by shifting arbitrarily small amount of flow $\epsilon > 0$ from the former path to one of the latter paths such that the minimum lifetime of the latter path after the shift is still larger than the minimum lifetime over all nodes before the shift. This contradicts our assumption that the minimum lifetime over all nodes is maximized. □

3.1 Algorithm Description

The algorithm FR tries to balance the minimum lifetime of each flow path by using an iterative feasible descent method. The main idea of FR is to redirect a portion of the current flow at each node i in a way that the minimum lifetime over all nodes will increase so that the resulting flow to the destination will eventually have the same lifetime in all paths, which satisfies the necessary optimality condition stated in Theorem 1.

In the following, we describe the implementation of FR. Let's use an imaginary super source node s where $S_s = O$, and an imaginary super sink node $\tilde{d}$ where $\tilde{d} \in S_d$ for all $d \in D$. Let $e_{so} = 0$ for all $o \in O$ and $e_{d\tilde{d}} = 0$ for all $d \in D$, and let E_s and $E_{\tilde{d}}$ be any positive constant. Let the initial flow be such that $q_{so} = Q_o$ and from o to $\tilde{d}$ the minimum transmitted energy path is used with a flow value Q_o for each $o \in O$. Note, however, that any feasible flow may be used as the initial flow. Each node $i \in N - D$ redirects its outgoing flow by subtracting ϵ_i from the flow of a certain path to the destination and by adding the same amount to the flow of another path to the destination. The steps to be taken at each node $i \in N - D$ are: i) Determine the from which path to which path; ii) Calculate ϵ_i, the amount of redirection; iii) Redirect the flow properly

Let P_i be the set of all paths from node i to the destination node $\tilde{d}$. For a path $p \in P_i$, define the path length $L_p(\mathrm{q})$ as a vector whose elements are the lifetimes of all the nodes in the path. For example, if path $p \in P_i$ starting from node i traverses nodes $j_1, j_2, \cdots, j_m$ where $j_m \in D$, then $L_p(\mathrm{q}) = [T_i(\mathrm{q}), T_{j_1}(\mathrm{q}), T_{j_2}(\mathrm{q}), \cdots, T_{j_{m-1}}(\mathrm{q})]$. The length of path p, $L_p(\mathrm{q})$ is said to be longer than the length of path p', $L_{p'}(\mathrm{q})$ if the smallest element of $L_p(\mathrm{q})$ is larger than that of $L_{p'}(\mathrm{q})$. In case they are the same, the next smallest elements are compared, and so on. Using this so-called lexicographical ordering, the longest length path between any two nodes can be defined. We modify the distance comparison part of the Bellman-Ford algorithm [2] to obtain the longest length paths distributively. Let's denote the longest length path from node i to the destination node $\tilde{d}$ as $l(i)$.

The first step of the algorithm at each node i is described in more detail. Let's denote the next hop nodes by g and t where we would like to redirect the flow q_{ig} to the flow q_{it}, where $g \in S_i$ and $t \in S_i$ are to be carefully chosen among the neighboring nodes of node i.

First of all, among the downstream neighboring nodes of node i, find the node whose longest path length to the destination is the smallest, and call it h, i.e.,

$$h = \underset{j\,:\,j \in S_i,\; q_{ij} > 0}{\mathrm{argMIN}} \; L_{l(j)}(\mathrm{q}), \tag{12}$$

where MIN denotes the minimum in the lexicographical ordering.

Now, depending on whether the lifetime of node i, $T_i(\mathrm{q})$ is larger than the minimum lifetime of this path, two different measures are taken.

If $T_i(\mathrm{q}) \leq \min[L_{l(h)}(\mathrm{q})]$ then we would like to increase the lifetime of node i. This can be achieved if we redirect the flow at node i from the direction where the required transmission energy per bit is the largest to the direction whose longest path length to the destination is the largest among all neighboring nodes of i with smaller transmission energy per information unit. That is,

$$g = \underset{j\,:\,j\in S_i,\; q_{ij}>0}{\operatorname{argmax}} \; e_{ij}, \tag{13}$$

and

$$t = \underset{j\,:\,j\in S_i,\; e_{ij}<e_{ig}}{\operatorname{argMAX}} \; L_{l(j)}(\mathrm{q}), \tag{14}$$

where MAX is the maximum in the lexicographical ordering. Note that $l(t)$ doesn't have to be the largest longest path, it suffices to have less energy consumption per bit transmission. In our algorithm, we alternately choose t to be the one with the minimum energy consumption per bit transmission, i.e.,

$$t = \underset{j\,:\,j\in S_i}{\operatorname{argmin}} \; e_{ij}. \tag{15}$$

On the other hand, if $T_i(\mathrm{q}) > \min[L_{l(h)}(\mathrm{q})]$ then we would like to increase the length of the smallest longest length path. This can be achieved if we redirect the flow from the smallest longest length path to the largest longest length path. That is,

$$g = h, \tag{16}$$

and

$$t = \underset{j\in S_i}{\operatorname{argMAX}} \; L_{l(j)}(\mathrm{q}). \tag{17}$$

Note that $l(t)$ doesn't have to be the largest longest path, it suffices to have larger minimum lifetime than that of the smallest longest path $l(h)$. In our algorithm, we alternately choose t to be the one with the minimum energy consumption per bit transmission.

$$t = \underset{j\,:\,j\in S_i, min[L_{l(j)}(\mathrm{q})]>min[L_{l(h)}(\mathrm{q})]}{\operatorname{argmin}} \; e_{ij}. \tag{18}$$

The second step of the algorithm at each node i is described in more detail. The constraints that ϵ_i should meet are as follows. First, it should be less than or equal to q_{ig} since no one can give what doesn't exist, i.e.,

$$\epsilon_i \leq q_{ig}. \tag{19}$$

Furthermore, none of the lifetimes should become smaller than the currently minimum lifetime of the subnetwork consisting of node i and all downstream nodes since this will lead us to the opposite direction to that of our objective. By this, we guarantee the monotonic convergence of the algorithm. If $T_i(\mathrm{q}) \leq \min[L_{l(h)}(\mathrm{q})]$ then none of the lifetimes in the path $l(t)$ should become smaller than $T_i(\mathrm{q})$, i.e.,

$$\frac{1}{T_j(\mathrm{q})} + \frac{e_{jk}\epsilon_i}{E_j} \leq \frac{1}{T_i(\mathrm{q})}, \tag{20}$$

for any link (j,k) in the path $l(t)$. On the other hand if $T_i(\mathrm{q}) > \min[L_{l(h)}(\mathrm{q})]$ then we need to consider two things. First, none of the lifetimes in the path $l(t)$ should become smaller than the minimum lifetime of the path $l(g)$, i.e.,

$$\frac{1}{T_j(\mathrm{q})} + \frac{e_{jk}\epsilon_i}{E_j} \leq \frac{1}{\min[L_{l(g)}(\mathrm{q})]}, \tag{21}$$

for any link (j,k) in the path $l(t)$. Second, if $e_{it} > e_{ig}$ then the lifetime of node i may decrease due to the redirection, but it should not become smaller than the minimum lifetime of the path $l(g)$, i.e.,

$$\frac{1}{T_i(\mathrm{q})} + \frac{(e_{it} - e_{ig})\epsilon_i}{E_i} \leq \frac{1}{\min[L_{l(g)}(\mathrm{q})]}. \tag{22}$$

Finally, the value of ϵ_i should be chosen among the values that meet all the constraints stated above. We could either use the maximum ϵ_i that meets all the constraints or just a fraction of it. To avoid possible oscillations and for faster convergence, we choose the half of the maximum ϵ_i that meets all the constraints except for (19).

The third step of the algorithm at each node i is described in more detail. Adding ϵ_i to the path $l(t)$ is easy. We can simply add it to each link flows in the path. However, subtracting ϵ_i from the path $l(g)$ is not so easy since there may be some links in the path whose flow is less than ϵ_i. We start to subtract ϵ_i from q_{ig}, which is safe because of (19). Suppose that ϵ_i is larger than the flow q_{jk} at some link (j,k) in the path $l(g)$ where we want to subtract ϵ_i from. We use a recursive algorithm in this case. First, we subtract q_{jk} from q_{jk} and then continue subtracting this amount along the path $l(k)$, the longest length path from k to d. Second, we subtract the remaining $\epsilon_i - q_{jk}$ from the smallest longest path that starts from a downstream neighbor h' of node j, i.e.,

$$h' = \underset{x\,:\,x \in S_j,\ q_{jx} > 0}{\mathrm{argMIN}} \ L_{l(x)}(\mathrm{q}). \tag{23}$$

If $\epsilon_i - q_{jk}$ is still larger than the flow $q_{jh'}$ then we successively try the next smallest longest path that starts from a downstream neighbor of node j, and so on.

Algorithm FR doesn't always converge to the global optimum. An example showing the convergence to the local optimum is presented in the following. While Figure 3 (a) is the global optimum, the algorithm may converge to Figure 3 (b) depending on the initial flow. The numbers correspond to the case when $\delta = 0.01$ in the figure, and again the information generation rate at the origin is assumed to be $Q_o = 1$. While the global optimal system lifetime is $T_{sys}^{opt} \approx 1.99$, the local optimum by FR is $T_{sys}^{FR} \approx 1.01$. In this case the ratio between the two is given by $R = 0.5075$, where the ratio

$$R = \frac{T_{sys}^{X}}{T_{sys}^{opt}} \tag{24}$$

will be used as the performance measure of algorithm X.

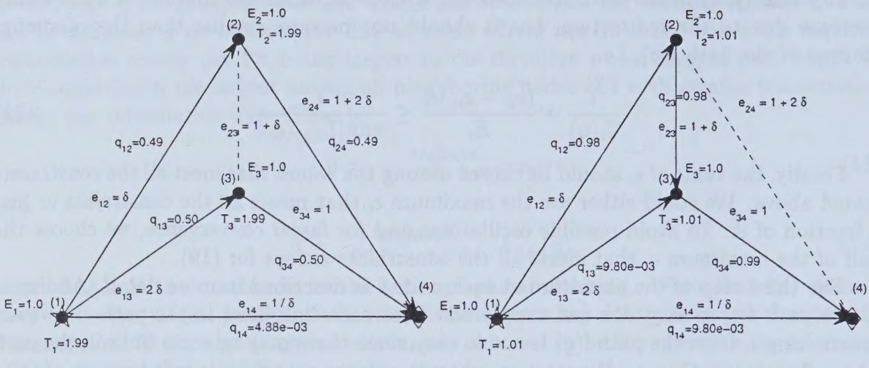
We could observe that as δ approaches zero the ratio between the local optimum and the global optimum approaches 0.5. We are not sure if this is indeed the worst case of FR as of yet. However, we haven't found any simulation result with $R < 0.5$.

4 Maximum Residual Energy Path Routing

In this section, we propose another algorithm based on Theorem 1 called the maximum residual energy path (MREP) routing algorithm.

Let P_i be the set of all paths from node i to the destination node $\tilde{d}$. For a path $p \in P_i$, define the path length L_p as a vector whose elements are the reciprocal of the residual energy for each link in the path after the route has been used by a unit flow. We assume that the routing path is calculated for each unit flow. The element of L_p for link (j,k) is

$$\frac{1}{\underline{\mathrm{E}}_j - e_{jk}\hat{u}}, \tag{25}$$



(a) (b)

Figure 3: An example showing local optimum convergence : (a) global optimum with $T_{sys}^{opt} \approx 1.9901$, (b) local optimum with $T_{sys}^{FR} \approx 1.0099$ when $\delta = 0.01$.

where $\underline{E}_j$ is the residual energy at node j and $\hat{u}$ is a unit flow. By using the lexicographical ordering in this case by comparing the largest elements first and so on, shortest path from each node i to the destination can be obtained using a slightly modified version of the distributed Bellman-Ford algorithm. A unit flow is transmitted via the shortest path.

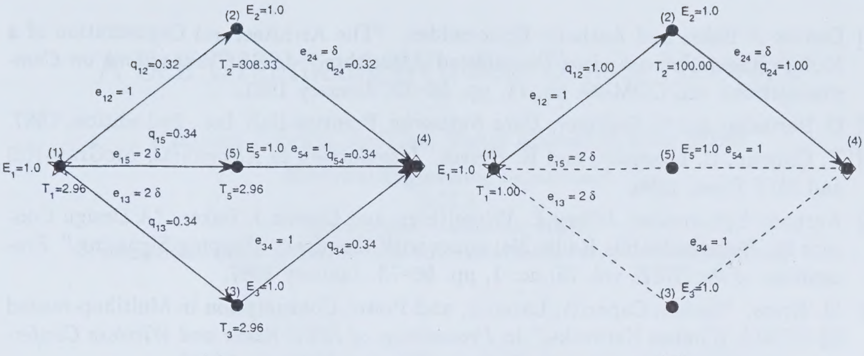
5 Performance Comparison through Simulation

In this section, random graphs are generated to evaluate the performances of the algorithms. The algorithms are compared with the conventional minimum transmitted energy (MTE) algorithm in order to see how much we gain in terms of the system lifetime by adopting the new problem formulation. MTE algorithm is a shortest path algorithm where the path length is the sum of energy expenditures per bit transmission over each link in the path.

We can find a pathologic example where the ratio between the maximum lifetime of the system obtained by MTE and the true optimum is close to zero. Consider Figure 4 for example where node 1 is the origin and node 4 is the destination. Assume that the information generation rate at the origin is $Q_o = 1$. While Figure 4 (a) shows the true optimum where $T_{sys}^{opt} \approx 2.96$, Figure 4 (b) shows that the maximum lifetime of the system obtained by MTE is $T_{sys}^{MTE} = 1.00$ when $\delta = 0.01$. This corresponds to $R \approx 0.3378$. As δ goes to zero the ratio R approaches 1/3. By adding more nodes with the distance of 2δ from node 1 and with the distance of 1 to node 4 like node 5, the ratio R becomes approximately inversely proportional to the number of such nodes added.

Random graphs are generated according to the following description. Let there be 20 nodes randomly distributed in a square of size 5 by 5 among which 5 origin nodes and 2 destination nodes are randomly chosen. Let each node i have initial energy $E_i = 1$ and the information generation rate at each origin node $o \in O$ is $Q_o = 1$. We assume that the transmission range of each node is limited by 2.5. The energy expenditure per bit transmission from node i to j is given by

$$e_{ij} = \left\{ \max\left(0.01, \frac{d_{ij}}{2.5}\right) \right\}^4, \tag{26}$$


(2) E_2=1.0
q_{12}=0.32
T_2=308.33
e_{24} = δ
q_{24}=0.32
e_{12} = 1
q_{15}=0.34
(1)
e_{15} = 2 δ
(5) E_5=1.0 e_{54} = 1
q_{54}=0.34
(4)
E_1=1.0
T_1=2.96
e_{13} = 2 δ
T_5=2.96
q_{13}=0.34
e_{34} = 1
q_{34}=0.34
(3) E_3=1.0
T_3=2.96
(2) E_2=1.0
q_{12}=1.00
T_2=100.00
e_{24} = δ
q_{24}=1.00
e_{12} = 1
(1)
e_{15} = 2 δ
(5) E_5=1.0 e_{54} = 1
(4)
E_1=1.0
T_1=1.00
e_{13} = 2 δ
e_{34} = 1
(3) E_3=1.0

(a) (b)

Figure 4: An example showing MTE's arbitrarily poor performance : (a) true optimum with $T_{sysx}^{opt} \approx 2.96$, (b) system lifetime obtained by MTE, $T_{sys}^{MTE} = 1.00$ when $\delta = 0.01$.

where $d_{ij} \leq 2.5$ is the distance between nodes i and j. Note that there may be cases where no path is available between the origin and the destination. We simply discard those cases to assume the connectivity.

The performances of the MTE, FR, and MREP are compared in Table 1.

Table 1: The performance comparison of the three algorithms.

R	MTE	FR	MREP
average	0.7519	0.9538	0.9607
worst case	0.2160	0.6567	0.8349
over 0.9	37 %	84 %	89 %

On the average, FR and MREP were both close to optimum with comparable performances, while MTE was not as good as the two. While the mean of the ratio R of MTE was about 0.75, that of FR and MREP were about 0.96. The ratio R of FR and MREP were over 90 % of the true optimum in 84 % and 89 % of the case respectively, while that of MTE was so in only 37 % of the case. The worst case performance of MTE was 0.2160 of the optimum. Note that the performance of MTE was shown to be arbitrarily bad. The worst case of FR was 0.6567, and the worst case of MREP was 0.8349 which is better than that of FR.

6 Conclusion

A routing problem in a power-controlled wireless ad-hoc network of static nodes was formulated. We assumed that the limited battery energy is the single most important resource and hence formulated the problem as to maximize the battery lifetime of the system for a set of given information generation rates at the origin nodes. In order to maximize the lifetime, the traffic is routed such that the energy consumption is balanced among the nodes in proportion to their energy reserves, instead of routing to minimize the absolute consumed power. We proposed the flow redirection algorithm and the maximum residual energy path routing algorithm that are local and amenable to distributed implementation with close to optimal performance most of the time.

References

[1] Dennis J. Baker and Anthony Ephremides, "The Architectural Organization of a Mobile Radio Network via a Distributed Algorithm," *IEEE Transactions on Communications*, vol. COM-29, no. 11, pp. 56–73, January 1981.

[2] D. Bertsekas and R. Gallager, *Data Networks*, Prentice-Hall, Inc., 2nd edition, 1987.

[3] T. Cormen, C. Leiserson, and R. Rivest, *Introduction to Algorithms*, McGraw-Hill and MIT Press, 1990.

[4] Anthony Ephremides, Jeffrey E. Wieselthier, and Dennis J. Baker, "A Design Concept for Reliable Mobile Radio Networks with Frequency Hopping Signaling," *Proceedings of the IEEE*, vol. 75, no. 1, pp. 56–73, January 1987.

[5] M. Ettus, "System Capacity, Latency, and Power Consumption in Multihop-routed SS-CDMA Wireless Networks," in *Proceedings of IEEE Radio and Wireless Conference (RAWCON) 98*, Colorado Springs, CO, Aug 1998, pp. 55–58.

[6] R.G. Gallager, P.A. Humblet, and P.M. Spira, "A Distributed Algorithm for Minimum Weight Spanning Trees," Tech. Rep. LIDS-P-906-A, Lab. Inform. Decision Syst., Massachusetts Inst. of Technol., Cambridge, MA, October 1979.

[7] D. Johnson and D. Maltz, "Dynamic Source Routing in Ad hoc Wireless Networks," *Mobile Computing*, 1996.

[8] Teresa H. Meng and Volkan Rodoplu, "Distributed Network Protocols for Wireless Communication," in *Proceedings of the 1998 IEEE International Symposium on Circuits and Systems, ISCAS'98*, Monterey, CA, June 1998, vol. 4, pp. 600–603.

[9] A. Michail and A. Ephremides, "A Distributed Routing Algorithm for Supporting Connection-oriented Service in Wireless Networks with Time-varying Connectivity," in *Proceedings Third IEEE Symposium on Computers and Communications, ISCC'98*, Athens, Greece, June 1998, pp. 587–591.

[10] S. Murthy and J.J. Garcia-Luna-Aceves, "An Efficient Routing Protocol for Wireless Networks," *ACM Mobile Networks and Applications Journal, Special Issue on Routing in Mobile Communication Networks*, 1996.

[11] S. G. Nash and A. Sofer, *Linear and Nonlinear Programming*, McGraw-Hill, 1996.

[12] Vincent D. Park and M. Scott Corson, "A Highly Distributed Routing Algorithm for Mobile Wireless Networks," in *Proc. IEEE INFOCOM'97*, Kobe, Japan, 1997.

[13] C. Perkins and P. Bhagwat, "Highly Dynamic Destination-sequenced Distance Vector Routing (DSDV) for Mobile Computers," in *ACM SIGCOMM*, October 1994.

[14] Volkan Rodoplu and Teresa H. Meng, "Minimum Energy Mobile Wireless Networks," in *Proceedings of the 1998 IEEE International Conference on Communications, ICC'98*, Atlanta, GA, June 1998, vol. 3, pp. 1633–1639.

[15] Timothy Shepard, "Decentralized Channel Management in Scalable Multihop Spread Spectrum Packet Radio Networks," Tech. Rep. MIT/LCS/TR-670, Massachusetts Institute of Technology Laboratory for Computer Science, July 1995.

[16] S. Singh, M. Woo, and C.S. Raghavendra, "Power-aware Routing in Mobile Ad Hoc Networks," in *Proceedings of Fourth Annual ACM/IEEE International Conference on Mobile Computing and Networking*, Dallas, TX, Oct 1998, pp. 181–190.

A Self Organizing Wireless Sensor Network*

Katayoun Sohrabi, Jay Gao, Vishal Ailawadhi, Greg Pottie

Electrical Engineering Department
UCLA
Los Angeles, California, 90095
sohrabi@ee.ucla.edu gao@ee.ucla.edu vishal@ucla.edu pottie@icsl.ucla.edu

Abstract

We describe a protocol suite for self-organization of sensor networks, in which the number and topology of the sensor nodes are a priori unknown, and no timing has been made available. The scalable protocol very quickly establishes a TDMA and frequency assignment schedule, and then proceeds to establish energy-efficient routing. Simulation results will be demonstrated. We are in the process of porting the protocol to our network of 40 sensor nodes, and incorporating limited mobility features.

1 Introduction

With continued advances in integrated circuits technology, it is now possible to build very compact and low-cost devices that include sensing, signal processing, and wireless communications. Networks of such devices can then be constructed for such diverse purposes as security, machine monitoring, remote medical patient care, inventory control, and indeed many others in which a low cost connection between networks and the physical world can be made. It is imperative in such circumstances that the sensor network be self-organizing and energy efficient, since with a large number of nodes the installation cost would otherwise be large, and the energy storage system (hardware and maintenance) could also place a floor on cost. Under a joint research program of UCLA and the Rockwell Science Center, prototype platforms have been constructed which can be used for both gathering sensor data in support of target classification and location algorithm development, and to test the self-organizing network protocols to be described below. In this paper we describe an autonomous and scalable boot-up procedure, an energy efficient routing algorithm, and a protocol to allow interaction of pedestrians with the static sensor network. All of these operate without the need for manual configuration, prior information on the number or identities of the network elements, or synchronization. Section 2 describes the self-organizing algorithm and MAC, Section 3 describes the routing algorithm, Section 4 gives the extension to mobility, and Section 5 presents our conclusions.

---

*This research is supported by DARPA Contract number F04701-97-C-0010

2 Link Layer Issues

The sensor network must be able to self organize at the lowest level, i.e. the member nodes must be able to form a connected multi-hop network dynamically and autonomously. The key design issues are the available bandwidth (which is large, compared to the per hop data rate) and available energy (which is small). The BOOTUP algorithm conserves energy by reducing the number of messages passed to build the network. This is done by combining the connectivity discovery phase with the link assignment phase and removing the requirement for network-wide slot synchronization. The resulting solution is a distributed and scalable algorithm which builds a flat hierarchy. The final product of this organization algorithm will be a connected multi-hop network, where each node follows a TDMA-like time schedule.

2.1 Self Organization Problem

The BOOTUP algorithm provides a means for a random collection of nodes to determine, collectively, local versions of the connectivity matrix of the network, and to establish links based on this information. The nodes do not have any prior information about the time reference, or location of other nodes or total number of nodes when the algorithm starts.

Assume there are R orthogonal channels available to our network. (e.g. through some combination of time, frequency, or code division). Consider a network of N nodes, with connectivity matrix $C = [c_{ij}]$. A link l_{ij} exists from i to j, if j is able to receive signal from i and $c_{ij} = 1$. Otherwise $c_{ij} = 0$. Let this network have a total of L links, numbered from 1 to L . Two links l_1, from node i to node j , and l_2, from node k to node m , are defined to be interfering if at least one of the members of the following set c_{kj}, c_{im} is equal to one. Now an interference free organization of the network will be a mapping, such that in the entire network no two interfering links have the same channel r assigned to them. The size of this organization is the total number of distinct channels assigned. Finding a mapping where the total number of assigned channels is minimum is an NP complete problem for general topologies. Fortunately, we only require an interference-free mapping, regardless of its size. The important constraint is that this mapping must be found using minimal energy. Since the major energy cost is due to communicating messages between nodes, our algorithm must find ways to reduce the number of passed messages.

To summarize, the self organization procedure must form a connected multi-hop network, starting with no connectivity information or timing reference. The algorithm must also save energy. In order to do this, the algorithm enables nodes to find their neighbors by means of exchanges of a limited number of messages over the air. The messages enable nodes to change local information, and assign channels to the discovered link in a distributed fashion.

2.2 Channel Access Mechanism

Before description of the BOOTUP algorithm is given, the channel access mechanism for the sensor network must be described. In the sensor network nodes communicate intermittently, and their radios, which are a major energy consumer, need not be turned on at all time. This leads to a TDMA-like channel access mechanism for the sensor network. Unfortunately, generic TDMA systems are synchronous systems, where all the nodes are slot synchronized (a costly endeavor). However, since a node only needs to know the transmission and reception epochs of its neighbors, and since these coordinates are determined at the time a link is formed, there is no need to form network wide synchronization, nor for a single TDMA schedule. Thus nodes do not reckon their frame structure on a single frame epoch. Also the internal structure of this frame structure, which we call a "SUPER FRAME", will be different for each node. Figure 1 illustrates the concept of the "SUPER FRAME".

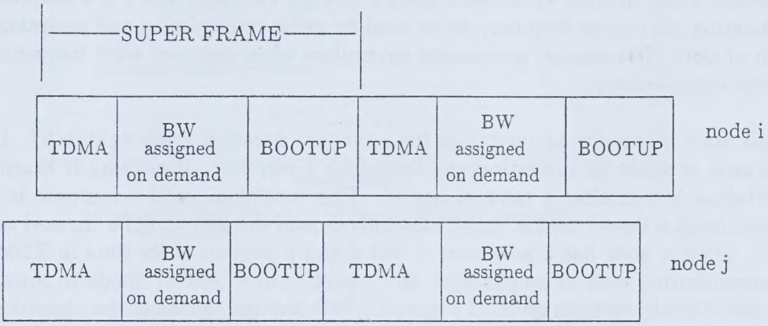


Figure 1: SUPER FRAME structure.

2.3 BOOTUP Protocol

In many classical organization algorithms two passes are needed over the entire network. In the first, all nodes follow a certain pre-assigned order, to broadcast their existence. By the end of the first pass, all nodes have a good idea about their local connectivity. The second pass enables the nodes to make decisions about the allocation of channels to links, and discover and remove interfering assignments. All the nodes must be listening to the channel for entire duration of both passes.

In our algorithm, the nodes turn off their transceivers, and only listen and transmit on the channel for a fraction of the duration of the time the self-organization procedure runs. There is also no a priori time synchronization. Our solution enables each node to find its radio neighbors one at a time. At the same time a link is formed, a channel is mutually agreed upon (based on the connectivity information of the two nodes involved) and assigned to the new link. This approach may result in less than optimum bandwidth utilization. However, since bandwidth is not a premium, we will accept this in exchange for energy savings that result from passing very few messages.

All nodes in the network run identical copies of the BOOTUP algorithm upon their respective wake-up times (which we assume are randomly distributed). The algorithm requires each node to transmit and wait to receive in a certain sequence a set of four message types called *TYPE1, TYPE2, TYPE3* , and *TYPE4* . *TYPE1* message is an invitation message, and the rest comprise a three message handshake. By the end of a successful handshaking transaction, the two nodes involved will have information about the TDMA schedule assignments of each other (effectively the one-hop away connectivity information), and also will have agreed upon a pair of time slots and an operational frequency band as the channel assigned to the link between them.

We assume each radio transceiver is able to tune to different frequencies while engaging in normal communications with neighbors (i.e. during the TDMA frame). However the entire BOOTUP algorithm takes place on a single fixed frequency, known to all the network nodes. A channel here is an ordered triplet *(i,j,k)* where *i* and *j* are slot orders, and indicate a pair of slots within each node's SUPER FRAME, and *k* is a frequency index, indicating the carrier frequency to be used for radio transmission and reception over the pair of slots. The channel assignment determines when and over what frequency the two nodes communicate.

Each node listens for an invitation for a random duration upon waking up. If nothing is heard, it sends an invitation and listens for a response. If nothing is heard, another invitation is sent after a random period. This continues, until a response is heard, at which point a three message handshake follows, and the pair assign a channel to this new link. Once a node has a neighbor, it will spend a portion of its time in TDMA mode, communicating with its neighbor(s), and a portion in BOOTUP mode to search for new nodes. The algorithm is given in Figure 2 . Detailed description of the algorithm is given in [7].

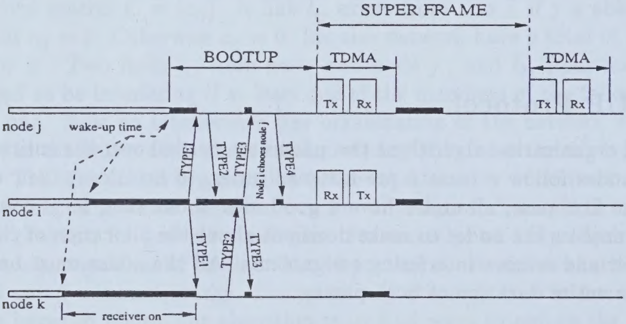


Figure 2: BOOTUP algorithm.

2.4 Performance of the BOOTUP algorithm

The algorithm is implemented in the PARSEC environment [8]. Its performance in terms of the amount of time and energy required to connect the network was investigated. Figure 3 gives the average performance for the time until the network becomes connected and also the duration the radios are powered on. Note that if the radio burns the same amount of energy while transmitting and receiving, then the duration of radio power up time will be good indication of the consumed energy. For purposes of comparison, the duration of the radio power-up time, for a single run of the Linked Cluster Algorithm,

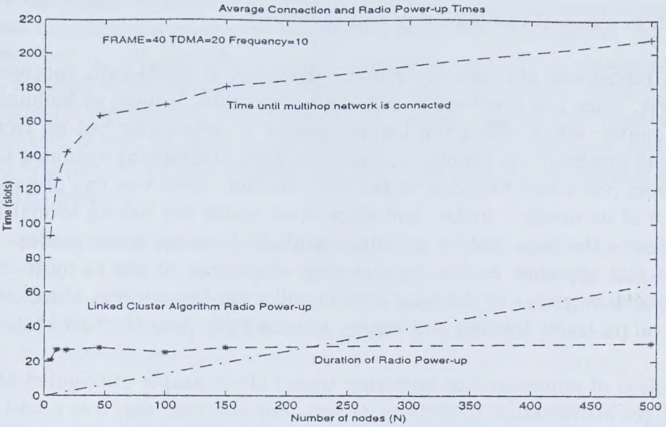


Figure 3: Performance of the BOOTUP algorithm.

LCA [6] , is also included in this graph. Note that, the LCA assumes node synchronization at the slot level. It also requires knowledge of the total number of nodes, *N*, as well as an agreement between all the nodes about the location of transition time of node 0, i.e. frame synchronization. These are all requirements which do not have to be met in the BOOTUP algorithm.
It is clear that the algorithm is scalable. Also note that it connects the network quickly. The algorithm is able to organize the network without any node having prior knowledge of other node timing reference, as can be seen in Figure 3. We have been able to connect networks of up to 500 nodes with this algorithm. The energy saving advantage of the algorithm, is evident for large numbers of nodes, as can be seen from Figure 3.

3 Multi-hop Routing Issues

Energy management will be the primary routing consideration because it has a direct impact on network topology. Although most applications involved stationary sensor nodes, battery depletion can introduce some unpredictable changes in network connectivity. Since radio communication is one of the most energy consuming operations, topological change patterns in the network can be highly correlated to the radio traffic loading.

Sensor network traffic is highly asymmetric. Traffic flows in two primary directions: (1) **user-to-sensor**, which consists of command messages issued by the network user, through a base station or long range unit, to subsets of sensor nodes as query for status reports or trigger for some network functionality, and (2) **sensor-to-user**, which includes reports for target detection, activity, cooperative sensor function results, or in rare cases, even raw sensor data. Since the sensor-to-user flow dominates, a corresponding asymmetry in energy consumption is created, making the immediate neighborhood of the base station more susceptible to energy depletion. Therefore, an important task in traffic management is to maintain the network connectivity near the base station. Also, most sensor-to-user messages are prioritized based on the nature of its payload, which require different QOS in terms of routing delay and reliability.

To meet the robustness and priority routing demands, a multi-path approach is pursued. However, since low overhead is still very important, instead of building node or link-disjoint routes, which offers the highest degree of robustness but an equally high cost in terms of overhead, we simply construct multiple overlapping spanning trees, each originating from one 1-hop neighbor of the base station. Each tree can provide a single route for each of its member nodes, and since most nodes will belong to multiple trees, multiple routes to the base station are made available to most sensor nodes. A hidden advantage to this approach is that by choosing which tree to use to route, each node can control which neighbor of the base station will relay the packets, therefore, gaining indirect control on traffic loading and energy consumption near the base station.

The construction of minimum-hop spanning trees follows simple distributed algorithms, with some slight modification to restrict its size. For any tree, say, tree i, and one of its member nodes, say, node A, two parameters will be computed for routing purposes: (1) a path metric, M_i, which is the sum of additive QOS parameter on all links and nodes between node A and the base station on tree i, and (2) a tree life estimate, L_i, which measures the estimated number of radio transmissions that can be supported on tree i before energy depletion occurs.

By using additive metrics, we assume that larger metric values correspond to lower QOS. Another related parameter would be w_i, which is a weight coefficient assigned to the ith packet generated by node A. Let p_i denote the tree selected by node A to route the ith packet. For performance evaluation, we assume node A belongs to m spanning trees, and define the average weighted metric to be:

$$\bar{M} = \frac{1}{N(m)} \sum_{i=1}^{N(m)} M_{p_i} w_i \tag{1}$$

where $N(m) = \sum_{j=1}^{m} L_j$ is the estimated number of radio transmissions that the combined resource of the m trees can support. $\bar{M}$ also represents the average weighted QOS provided to each packet. We can then define a **resource matrix**:

$$\mathbf{r} = \begin{bmatrix} r_1 \\ r_2 \end{bmatrix} = \begin{bmatrix} M_1 \ldots M_1 & M_2 \ldots M_2 & \ldots & M_m \ldots M_m \\ \underbrace{1 \ldots\ldots 1}_{L_1} & \underbrace{2 \ldots\ldots 2}_{L_2} & \ldots & \underbrace{m \ldots\ldots m}_{L_m} \end{bmatrix} \tag{2}$$

Each column of $\mathbf{r}$ represents the capacity to route one packet from node A to the base station. Note that $\forall i, r_{1,i}$ is the metric value associated with $r_{2,i}$, the tree ID number. Let $q_i(p_i)$ be any permutation of $\{1, 2, 3, \ldots, N(m)\}$ such that $N(p_{i-1}) \leq q_i \leq N(p_i)$, so that it maps tree decision p_i onto r_i. Then, Eq (1) is equivalent to:

$$\bar{M} = \sum_{i=1}^{N(m)} w_i r_{1,q_i} \tag{3}$$

3.1 Sequential-Assignment Routing(SAR)

The goal of our routing algorithm is to minimize the quantity in Eq (3). Since the assignment of each packet to a path has to be made in real time, without the opportunity to observe the entire sequence of w_i which is random, the minimization problem belongs to the class of sequential stochastic assignment problems described in [4]. The optimal policy described in [4], though it can be translated easily from maximization to minimization, requires an iid assumption on w_i, knowledge of its distribution, and calculation of many integral expressions. Although the iid assumption is later removed in [5], the calculation still cannot be made energy efficient. So we only assumed having knowledge of the cdf, $F_W(w)$, of w_i and adopted the following suboptimal sequential assignment routing(SAR) policy:

> *For any node, say A, assume, without loss of generality, that* $M_1 \geq M_2 \geq \ldots \geq M_m$*, so that* $r_{2,j}$*'s are also in descending order. If by the best estimate, the network is determined to have resource capacity to route a total of N packets and that for the ith packet,* $w_i \in (F_W^{-1}(\frac{j-1}{N}), F_W^{-1}(\frac{j}{N})]$*, where* $1 \leq j \leq N$*, then the optimal tree assignment will be* $r_{1,j}$ *with metric value* $r_{2,j}$*.*

To maintain routing table integrity whenever failures occur, a local computation will be triggered to remove the invalid route. Also, path metrics and tree life estimates need to be kept on track with current network conditions because changes in the radio channel or additional energy consumption within the network may occur after the initial route computation. One way to accomplish this is to periodically trigger a route recomputation from the base station.

3.2 Simulation Results

We compare our algorithm to the classic Shortest Path algorithm, which will route all packets on the best possible path, regardless of priorities. A network with one base station and 37 randomly deployed sensors is simulated. We assume network traffic is generated by a subgroup of 5 sensors which detected extended target activities. Each member of the subgroup has equal probability of generating packets, and each packet has three possible priority settings, {low,medium,high}. Low and medium priority packets are assigned a weight coefficient of 0 and 1, respectively. Low priority traffic account for 30% of all traffic. Each link is assigned with one of two possible metric values to reflect different QOS. For the network simulated, a "good zone" is defined so that all links terminating in this zone have metric of 0; otherwise, the metric will be 30. Radios are assumed to operate at fixed power level and the per packet energy consumption is 10 energy units. Each node has metric of 1.0 and an identical power source with 10^6 units of energy. Global metric recomputation is initiated by the base station after every 5000 sensor packets were received. There are two major traffic parameters: p_h, the probability that a packet generated has high priority, and the w_h, the weight coefficient assigned to it. Higher p_h means more high priority packets will be sent through the system and their impact on the overall QOS will be significant. In our simulation, average weighted metric is compared for the SAR and the shortest path algorithm, under different high priority traffic settings.

In Figure 4, we compare the average metric for $p_h \in [0.05, 0.4]$. As expected, $\bar{M}$ increases when more high priority packets are generated. For $w_h = 50$, SAR is 38% to 40% better than shortest path; for $w_h = 10$, SAR is about 37.5% better. For both SAR and shortest path, the metric values grow roughly linearly with p_h. SAR algorithm is able to produce a lower metric because it preserves better paths by limiting usage, thus making them available for high priority packet. On the other hand, the shortest path algorithm makes no such distinction based on priorities. It assigns many low-priority packets to the best path, which degrades the performance in the long run.

Another parameter of interest is the global recomputation interval. Frequent recomputation can significantly reduce network capacity. Figure 4 also shows that for both algorithms, the average network capacity takes a 20% loss when the frequency of recomputation increases by one order of magnitude. However, no significant difference in network capacity can be observed between SAR and Shortest Path.

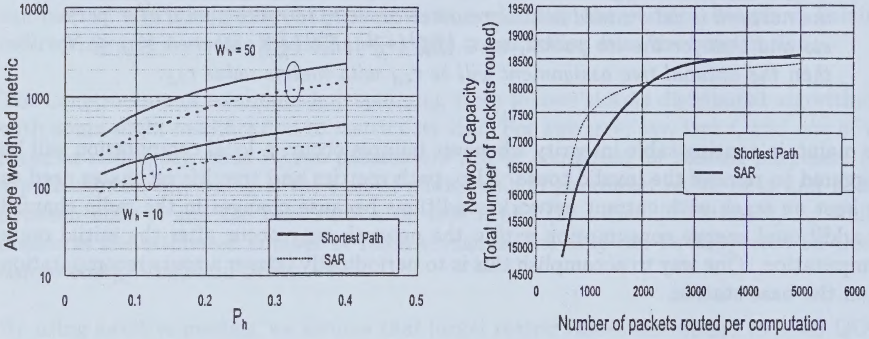


Figure 4: SAR & Shortest Path Comparison

4 Mobile MAC Algorithm

The goal of the mobile MAC algorithm is to enable connectivity between a limited number of mobile users and an existing sensor network. For example, a pedestrian may wish to down load data or issue instructions to the network via a handheld-device. This interaction should take place with the mobile having no prior knowledge of the network topology or channel assignments, and with minimal modification of the protocol of the sensor network. The protocol to be described below accomplishes these objectives by using the BOOTUP algorithm's invitation messages, and lies above the MAC of the stationary network. Since the mobile is more likely to have possibility of renewing its energy source, the protocol concentrates on conserving the energy of the stationary nodes, in contrast to cellular systems in which the stationary devices have the more reliable energy supply.

4.1 Algorithm Elements

The mobile assumes full responsibility for establishing and breaking connections. It makes use of the existing MAC, with the slight change that responses from mobiles are assigned higher priority, through preallocation of one slot per stationary node to handle mobile connections. This enables a more rapid build up and down of connections. The primary messages are:

Broadcast Invite (BI) The stationary node invites other nodes to join.

Mobile Invite (MI) The mobile responds to BI to request a connection.

Mobile Response (MR) The stationary node accepts or declines the MI request.

Mobile Disconnect (MD) The mobile informs the stationary node of a disconnect; no response is needed.

4.2 Protocol

Notice that in contrast to cellular hand-off protocols [2,3], very little information exchange is required. The mobile keeps a registry of nodes within communications range, which it periodically updates by means of interception of BI messages. Whether or not a connection is made depends on the status of its other connections, location and movement information inferred from the history of the connection and communications range registries, or other factors that may be important to a particular application. The stationary nodes have no responsibility for tracking any information regarding the location of the mobile node or the quality of the link.

The protocol amounts to how the mobile manages its two registries, which due to the heavy attenuation in ground to ground propagation will not be not very large. Here, by means of eavesdropping on BI messages the mobile begins filling its range registry with SNR's, node ID's, and other information on the local neighborhood. If the node is new, but the registry is full, a comparison is made based on the SNR to possibly knock out an old entry. Likewise, based on the SNRs in the registries a decision can be made to send out an MI to a listed node. For nodes that are already listed, reception of a BI message gives the opportunity to update link quality information, and decide whether a connection should be established or discontinued. An MI message, if issued, includes information regarding possible channel assignments, and will be issued only when strong SNRs are available to reduce the likelihood of frequent connects/disconnects with the same node.

Since in the application at hand the mobile node is likely to perform the role of an information sink, if the stationary node receiving an MI can establish a link (i.e., a time slot is available), then an MR message is issued. Otherwise, a short packet declining the invitation is sent. If the stationary node has between transmission of its BI and reception of the MI experienced some disaster, and no response is received, the mobile will after an appropriate period assume that no connection is possible. A mobile will likewise send an MD message to delete a connection if no traffic has arrived from a connected node for some period of time. MD messages will also be produced when SNR's on links drop below a threshold. The stationary node then deletes the mobile node from its registry of connections.

We have simulated connections of mobile users to randomly placed nodes with satisfactory results. The boot-up, routing, MAC, and mobile MAC protocols have been tested together in a unified simulation environment. We are working towards a more flexible assignment of slots in the TDMA structure for the mobile/stationary node interface, and on porting all the algorithms to our sensor node platforms.

5 Conclusion

We have described a protocol suite that provides self-organization for a randomly deployed sensor network, with the aim of establishing energy efficient communications. Remaining tasks include porting the code onto the sensor nodes, and tuning the parameters according to testing in the fieldAmsterdam, as well as development of specific network procedures to deal with cooperative processing requirements such as data fusion among sensor nodes and array beamforming.

References

[1] Pottie, G. J., "Wireless Sensor Networks," 1998 Information Theory Workshop, pp 139-40, 22-26 June 1998.

[2] Hansen, C. J. and Pottie, G. J., "A Distributed Access Algorithm for Cellular Radio Systems with Channel Partitioning," IEEE Transactions on Vehicular Technology, vol. 48, no. 1, pp 76-82, January 1999.

[3] Chen, J., et al., "A Comparison of MAC Protocols for Wireless Local Networks Based on Battery Power Consumption," IEEE INFOCOM 1998, vol. 1, pp 150-7.

[4] C. Derman, G. Lieberman, S. Ross, "A Sequential Stochastic Assignment Problem," Management Science, vol.18, No. 7, March, 1972

[5] D.P. Kennedy, "Optimal Sequential Assignment," Mathematics of Operations Research, vol. 11, No. 4, November 1986.

[6] D.J. Baker and A. Ephremides, "The Architectural Organization of a Mobile Radio Network Via A Distributed Algorithm", *IEEE Trans. Comm.* , vol. COM29, pp. 1694-1701, Nov. 1981.

[7] K. Sohrabi, G. Pottie, "Performance of a Novel Self-Organization Protocol for Wireless Ad-Hoc Sensor Networks", *IEEE Vehicular Technology Conference 1999* , , The Netherlands, September 1999.

[8] "Parsec: A Parallel Simulation Environment for Complex Systems," R. Bagrodia, R. Meyer, M. Takai, Y. Chen, X. Zeng, J. Martin, B. Park, H. Song; IEEE Computer, Vol.31(10), October 1998, pp. 77-85.

Blind Adaptive Noncoherent Multiuser Detection for Nonlinear Modulation*

Deepak Das, and Mahesh K. Varanasi Senior Member, IEEE

Dept. of Electrical and Computer Engineering
University of Colorado
Boulder, CO 80303
dasd@ucsu.colorado.edu varanasi@schof.colorado.edu

Abstract

Noncoherent multiuser detection for nonlinear modulation was studied in [1] where the idea of phase-independent noncoherent decorrelation was introduced and three post-decorrelative detectors were devised and analyzed. Their implementation however, requires the knowledge of the signature waveforms of all the users, which may be available only for centralized implementation. In this paper, we obtain a *blind* adaptive noncoherent decorrelative detector for nonlinear modulation that is suitable for distributed implementation with the knowledge of only the normalized signals of the desired user and the channel noise variance. This detector is based on the Robbins-Monro stochastic approximation method and does not require the overhead of training sequences. Two adaptive algorithms are developed, one guided by every signal individually, and the other, by the entire signal space of the desired user. Convergence of the algorithms can be shown in the Mean Squared Error (MSE) sense for both fixed step-size and time varying step-size versions of the two algorithms. For the time-varying step-size, the limiting MSE is equal to zero, and for the fixed step-size algorithms it is proportional to the step-size.

Index Terms: Noncoherent detection, nonlinear modulation, multiuser detection, stochastic approximation, blind receivers.

1 Introduction

Noncoherent multiuser detection for nonlinear modulation for the multiuser channel was introduced in [1] in the context of the additive white Gaussian noise (AWGN) channel. The need for noncoherent detection arises in applications where factors such as oscillator phase instability, mobility of the transmitter and/or receiver, and hence fading etc, cause rapid fluctuations of the carrier phase at the receiver that are difficult to track. A general nonlinear modulation technique that lends itself to noncoherent detection is Nonorthogonal Multipulse Modulation (NMM) where an M-ary symbol is transmitted by sending one of M possibly non-orthogonal and unequal-energy waveforms [1]. The usual equal-energy Orthogonal Multipulse Modulation (OMM) method is a special case of NMM. The distinct energies and the correlations between the signals in NMM can be

---

*This work was supported in part by NSF grant NCR-9725778 and by ARO grant DADD19-99-1-0291

either viewed as design parameters or they can arise as a result of a distorting channel in even systems that use OMM. The multiuser channel that we consider has several NMM based transmitters sending data simultaneously so that a superposition of several NMM waveforms plus additive white Gaussian noise is observed at the receiver. The multiple access technique employed is Correlated Waveform Multiple Access (CWMA) [1], where the signature waveforms employed by the different users can be correlated as well.

NMM can be viewed as psuedo-linear modulation in the expanded signal space of all the signals of all the users. This key idea allows the specification of an energy-and-phase-independent decorrelative operation in the expanded signal space to remove the multiple-access interference completely [1]. Three post-decorrelative detectors, namely, the optimum, the asymptotically optimum and the Generalized Likelihood Ratio Test (GLRT) based detectors were obtained in [1]. In this paper, we obtain a *blind* adaptive noncoherent post-decorrelative GLRT detector that is suitable for distributed implementation requiring just the normalized signature waveforms of the user of interest and the additive noise variance.

Past work in blind adaptive multiuser detection deals with coherent detection for linear modulation (where each user employs different levels of a single waveform). Several blind adaptive schemes for the two well-known decorrelating [2] and Minimum Mean Squared Error (MMSE) [3] detectors have been obtained. Among others, a blind adaptive algorithm that converged to the MMSE detector based on the minimization of the output energy was given in [4], and a blind adaptive decorrelator was presented in [5].

Of the two blind adaptive noncoherent decorrelators for NMM that are presented in this paper, the first is guided by every signal of a user individually, and is referred to as the *Single Signal based Blind Adaptive Noncoherent Detector* or SS-BAND. This can be seen as an extension of the blind adaptive decorrelator for linear modulation in [5]. The second adaptation is guided by the user's entire signal space and we refer to it as the *Group Signal-space Based Blind Adaptive Noncoherent Detector* or GS-BAND. The convergence to the deterministic decorrelator in the mean-squared error (MSE) sense, of a special Robbins-Monro [6] iteration-dependent step-size version of both the SS-BAND and the GS-BAND [1], follows along the lines of the classical proof of convergence of the Robbins-Monro stochastic approximation algorithm [6] [7]. The fixed step-size version of both the SS-BAND and the GS-BAND can be shown to converge to the deterministic decorrelator within a limiting MSE that is proportional to the step-size chosen (quasi-convergence).

The two blind adaptive noncoherent decorrelators can be used in conjunction with any of the three phase-independent post-decorrelative detectors obtained in [1]. The optimum and the asymptotically optimum detectors require knowledge of the signal energies of the user of interest, whereas the GLRT detector does not. Although suboptimal, the GLRT detector assumes no knowledge of the distribution of the complex amplitudes of the received signals, and is therefore applicable to channels other than the Gaussian channel with uniformly distributed phase that were studied in [1]. For instance, the GLRT detector can be applied to fading channels (such as Rayleigh, Rician, Nakagami, etc.) where the complex signal amplitudes and their distributions are unknown to the user.

2 System model

We consider a NMM-CWMA system, i.e., each user transmits one of M possibly nonorthogonal signals to send $\log_2 M$ bits of information. After passing through an

AWGN channel, the superposition of K signals arrives in symbol synchronism at the receiver with the following lowpass representation:

$$r(t) = \sum_{k=1}^{K} \sqrt{E_{ki_k}} e^{j\phi_{ki_k}} s_{ki_k}(t) + n(t), \quad t \in [0, T], \tag{1}$$

where n(t) is the AWGN with noise power spectral density (one-sided) of σ^2, and $s_{ki_k}(t), i_k \in \{1, \cdots, M\}$ are the M complex, equiprobable, and normalized (to unit-energy) signature signals of user k, time-limited to a period T. i_k is the symbol transmitted by user k and E_{ki_k} and ϕ_{ki_k} denote the energy and phase of the i_k^{th} signal of user k. It is implicitly assumed that the phases remain constant over each signalling interval. No knowledge of the complex amplitudes $E_{ki_k} e^{j\phi_{ki_k}}$ or their distributions is assumed to be available at the receiver.

The lowpass received signal can be projected onto an N-dimensional basis of orthonormal functions ($N \geq MK$) to yield the following vector representation for the received signal:

$$\mathbf{r} = \sum_{k=1}^{K} \sqrt{E_{ki_k}} e^{j\phi_{ki_k}} \mathbf{s}_{ki_k} + \mathbf{n} \tag{2}$$

where $\mathbf{s}_{ki_k}$ and $\mathbf{n}$ are the projections of the k^{th} user's i_k^{th} signal and the noise, respectively, on the orthonormal basis. For example, in a system where the signals have a direct-sequence spread-spectrum structure, the orthonormal functions could be the so-called chip waveforms. The noise vector $\mathbf{n}$ is a zero-mean white complex Gaussian random vector with covariance matrix $\sigma^2\mathbf{I}$, which we denote as $\mathbf{n}$ being $\eta(0, \sigma^2\mathbf{I})$. Representing user k's information by the random vector $\mathbf{b}_k^T \triangleq [b_{k1}, \cdots, b_{kM}]$ which equi-probably takes values in the set of M M-dimensional unit vectors, the N-length received vector admits the following psuedo-linear model

$$\mathbf{r} = \mathbf{SAb} + \mathbf{n}, \tag{3}$$

with $\mathbf{S} = [\mathbf{s}_{11} \cdots \mathbf{s}_{1M} \ \cdots \ \mathbf{s}_{K1} \cdots \mathbf{s}_{KM}] \in \mathbf{C}^{N \times MK}$, $\mathbf{b}^T = [\mathbf{b}_1^T, \ldots, \mathbf{b}_K^T]$, $\mathbf{A} = diag(\mathbf{A}_1, \ldots, \mathbf{A}_K) \in \mathbf{C}^{MK \times MK}$, and $\mathbf{A}_k = diag(\sqrt{E_{k1}} e^{j\phi_{k1}}, \cdots, \sqrt{E_{kM}} e^{j\phi_{kM}})$.

3 Noncoherent Decorrelative Detection

Let the decorrelator of interest (in the expanded signal space) be denoted by the $MK \times N$ phase and energy independent complex matrix transformation $\mathbf{C}$, such that

$$\mathbf{z} = \mathbf{C}^{\dagger}\mathbf{r} = \mathbf{Ab} + \gamma, \tag{4}$$

where γ is $\eta(0, \sigma^2\mathbf{C}^{\dagger}\mathbf{C})$. The decorrelator matrix for the k^{th} user, $\mathbf{C}_k$, is given by the k^{th} block of M columns of $\mathbf{C}$, where the decorrelator for the k^{th} user's m^{th} signal, i. e., the m^{th} column of $\mathbf{C}_k$, is denoted as $\mathbf{c}_{km}$. In order to decouple the multiuser channel into K single-user channels, we base the decision as to which signal is transmitted by user k only on the M-length decision statistic $\mathbf{z}_k$ given as

$$\mathbf{z}_k = \mathbf{C}_k^{\dagger}\mathbf{r} = \mathbf{A}_k\mathbf{b}_k + \gamma_k, \tag{5}$$

where γ_k is $\eta(0, \sigma^2\mathbf{C}_k^{\dagger}\mathbf{C}_k)$. Note that this is an information-lossy operation that enables us to obtain a decision statistic for user k that is free of the multiple-access interference but ignores the noise correlation with other users. Letting $\mathbf{P}_k \triangleq \left(\mathbf{C}_k^\dagger \mathbf{C}_k\right)^{-1}$, it follows from [1], that the GLRT based post-decorrelative detector for user k determines the decision for user k, $\hat{i}_k$, as follows:

$$\hat{i}_k = \arg\max_i \left\{ \frac{1}{P_k(i,i)} \left| \sum_{m=1}^{M} P_k(i,m) z_{km} \right|^2 \right\} . \tag{6}$$

We now need to specify the exact form of the decorrelating transformation $\mathbf{C}_k$ for user k, and note from (5) that $\mathbf{c}_{km}$ should satisfy the following condition:

$$\mathbf{S}^\dagger \mathbf{c}_{km} = \mathbf{e}_{M(k-1)+m} , \tag{7}$$

where $\mathbf{e}_i$ is the MK-length unit vector with a 1 in the i^{th} position. Since the MK signal vectors (the columns of $\mathbf{S}$) are assumed to be linearly independent, there could be more than one solution for $\mathbf{c}_{km}$. However, it is easy to show that the unique solution lying in the column space of $\mathbf{S}$ is the corresponding column of the psuedo-inverse of the signal matrix $\mathbf{S}$, given as

$$\mathbf{c}_{km}^* = \mathbf{S}\left(\mathbf{S}^\dagger \mathbf{S}\right)^{-1} \mathbf{e}_{M(k-1)+m} , \tag{8}$$

and this is the decorrelator we are interested in.

4 Blind Adaptive Noncoherent Decorrelation

Let us define the $MK \times MK$ matrix $\Theta \triangleq E\left(\mathbf{Abb}^T \mathbf{A}^\dagger\right)$ where E is the expectation operator. The $(i,j)^{th}$ $M \times M$ block, Θ_{ij}, of the matrix Θ is given as $\Theta_{ij} \triangleq E\left(\mathbf{A}_i \mathbf{b}_i \mathbf{b}_j^T \mathbf{A}_j^\dagger\right)$ where $i,j \in \{1, \cdots, K\}$. Based on the assumption that the phase associated with the signal sent by any user is independent of that associated with the signal sent by another user, it is easy to see that Θ_{ij} is the zero matrix for $i \neq j$. Moreover, equi-probable signalling implies that Θ_{ii} is a diagonal matrix whose m^{th} diagonal element is $\frac{E_{im}}{M}$. We can now write the following equation for the output correlation:

$$E\left(\mathbf{rr}^\dagger\right) = E\left[(\mathbf{SAb}+\mathbf{n})(\mathbf{SAb}+\mathbf{n})^\dagger\right] = \mathbf{S}\Theta\mathbf{S}^\dagger + \sigma^2 \mathbf{I} , \tag{9}$$

where we have assumed that the noise term is independent of the received complex amplitudes and the realizations of the M-ary information vector $\mathbf{b}$.

Starting with (7), we observe from the structure of the matrix Θ that

$$\mathbf{S}\Theta\mathbf{S}^\dagger \mathbf{c}_{km} = \frac{E_{km}}{M} \mathbf{s}_{km} . \tag{10}$$

It can be shown that if the MK signals are linearly independent, then all the solutions of (7) and (10) for $\mathbf{c}_{km}$ coincide. Therefore, any iterative algorithm that converges to a solution to (10) also yields a solution to (7).

Using the Robbins-Monro stochastic approximation method [7], we can devise the following fixed step-size stochastic adaptation algorithm to estimate a scaled version of $\mathbf{c}_{km}$ without the knowledge of the user energies:

$$\mathbf{c}_{km}(l+1) = \mathbf{c}_{km}(l) - \mu\left((\mathbf{r}(l+1)\mathbf{r}(l+1)^\dagger - \sigma^2 \mathbf{I})\mathbf{c}_{km}(l) - \mathbf{s}_{km} \right), \quad m = (1, \cdots, M). \tag{11}$$

Not using the knowledge of the user energies in (11), however, causes the estimate of the decorrelator to converge at best to a scaled version of the actual decorrelator, scaled by a factor of M/E_{km}. Using the scaled decorrelator, we obtain scaled versions of the decision statistic z_{km} and the terms $P_k(i,i)$ and $P_k(i,m)$ in (6). It is left to the reader to verify that the decision rule in (6) yields the same decisions when the scaled terms are used in place of the unscaled ones. Since only each signal $\mathbf{s}_{km}$ is used individually in the adaptation in (11), we refer to this algorithm as the Single Signal based Blind Adaptive Noncoherent Decorrelator(SS-BAND). If we replace the fixed step size μ by an iteration index dependent step size sequence μ_l (satisfying some conditions stated later), we obtain the Robbins-Monro stochastic adaptation algorithm to estimate $\mathbf{c}_{km}$.

Computing the decision statistics $\left\{z_{km}(l+1) = \mathbf{c}_{km}^{\dagger}(l)\mathbf{r}(l+1)\right\}$, $m = 1, \cdots, M$, the updation of the decorrelator estimate can be done efficiently with the following version of (11):

$$\mathbf{c}_{km}(l+1) = (1+\mu\sigma^2)\mathbf{c}_{km}(l) - \mu\left(\mathbf{r}(l+1)z_{km}^{\dagger}(l+1) - \mathbf{s}_{km}\right), \quad m = (1, \cdots, M). \quad (12)$$

The adaptation algorithm in (11) is a single signal based adaptation strategy where the correction term at each iteration depends only on the signal $\mathbf{s}_{km}$. A more appropriate adaptation scheme for NMM would require us to use the knowledge of all M signals, and this is what we shall investigate next. Without loss of generality, we will consider only the decorrelator for user 1. First, for notational simplicity, let us define the signal correlation matrix and its partitions as follows: $\mathbf{R} = \mathbf{S}^{\dagger}\mathbf{S}$, $\mathbf{S} = \left[\mathbf{S}_1 \mid \bar{\mathbf{S}}_1\right]$, $\mathbf{R}_A = \mathbf{S}_1^{\dagger}\mathbf{S}_1$, $\mathbf{R}_B = \mathbf{S}_1^{\dagger}\bar{\mathbf{S}}_1$ and $\mathbf{R}_C = \bar{\mathbf{S}}_1^{\dagger}\bar{\mathbf{S}}_1$, where clearly $\bar{\mathbf{S}}_1$ is composed of the signals of the remaining $K-1$ users. Let us also define two blocks along the diagonal of Θ, namely Θ_1 and $\bar{\Theta}_1$, where Θ_1 is the $M \times M$ northwest block and $\bar{\Theta}_1$ is the $M(K-1) \times M(K-1)$ southeast block.

In the rest of this paper, we will denote the projection operators for matrix Ψ as $\mathcal{P}_{\Psi} = \Psi\left(\Psi^{\dagger}\Psi\right)^{-1}\Psi^{\dagger}$ and $\mathcal{P}_{\Psi}^{\perp} = \mathbf{I} - \mathcal{P}_{\Psi}$. From the results on inversion of matrices [10], the conjugate transpose of the desired decorrelator matrix $\mathbf{C}_1^*$ (given by the first M columns of the psuedo-inverse of $\mathbf{S}$), can be written as:

$$(\mathbf{C}_1^*)^{\dagger} = \mathbf{R}_A^{-1}\left[\mathbf{I} - \mathbf{R}_B\left(\mathbf{R}_B^{\dagger}\mathbf{R}_A^{-1}\mathbf{R}_B - \mathbf{R}_C\right)^{-1}\mathbf{R}_B^{\dagger}\mathbf{R}_A^{-1} + \mathbf{R}_B\left(\mathbf{R}_B^{\dagger}\mathbf{R}_A^{-1}\mathbf{R}_B - \mathbf{R}_C\right)^{-1}\right]\bar{\mathbf{S}}_1^{\dagger}. \quad (13)$$

Defining the matrix $\mathbf{D}_1^{\dagger} \triangleq \mathbf{R}_A^{-1}\mathbf{R}_B\left(\mathbf{R}_B^{\dagger}\mathbf{R}_A^{-1}\mathbf{R}_B - \mathbf{R}_C\right)^{-1}\bar{\mathbf{S}}_1^{\dagger}$ and substituting in (13), for $\mathbf{R}_B$ and $\mathbf{R}_C$ in terms of $\mathbf{S}_1, \bar{\mathbf{S}}_1$, we obtain the following:

$$\mathbf{C}_1^* = \mathbf{S}_1\mathbf{R}_A^{-1} + \mathcal{P}_{\mathbf{S}_1}^{\perp}\mathbf{D}_1. \quad (14)$$

Note that the matrix $\mathbf{D}_1$ captures the interfering signal space information that user 1 has no access to in a completely distributed system. Consequently, this is the part of the decorrelator that we really need to adapt to. Denoting the m^{th} column of $\mathbf{D}_1$ as $\mathbf{d}_{1m}$, and the m^{th} column of $\mathbf{R}_A^{-1}$ as $\mathbf{q}_{1m}$ we obtain the following representation of the m^{th} decorrelator in terms of user 1's group of signals:

$$\mathbf{c}_{1m}^* = \mathbf{S}_1\mathbf{q}_{1m} + \mathcal{P}_{\mathbf{S}_1}^{\perp}\mathbf{d}_{1m}\,. \quad (15)$$

We remark here that the orthogonal decomposition of $\mathbf{c}_{1m}^*$ in (15) is the appropriate extension, to the *group* signal-space based approach, of a single signal based orthogonal decomposition, $\mathbf{c}_{1m}^{*} = \mathbf{s}_{1m} + \pi_{1m}$, $\mathbf{s}_{1m} \perp \pi_{1m}$, as suggested in [4] in the context of coherent MMSE detection for linear modulation.

In the case that user 1's signal subspace is orthogonal to the interfering signal subspace, we can exactly construct the corresponding decorrelators, $\mathbf{c}_{1m}^{*}$, with just the knowledge of $\mathbf{S}_1$, i. e., $\mathbf{C}_1^{*} = \mathbf{S}_1\mathbf{R}_A^{-1}$. However, user 1 may not know whether this is true of the signal set of its interferers and may still need to adaptively determine its decorrelator matrix.

Using (9) and (10) we obtain that

$$E\left[\mathcal{P}_{\mathbf{S}_1}^{\perp}\left(\mathbf{r}(l+1)\mathbf{r}(l+1)^{\dagger}-\sigma^2\mathbf{I}\right)\left(\mathbf{S}_1\mathbf{q}_{1m}+\mathcal{P}_{\mathbf{S}_1}^{\perp}\mathbf{d}_{1m}\right)\right]=\mathbf{0}\,, \tag{16}$$

where $\mathbf{0}$ is the N-length zero vector. As in (11), we can now devise the following fixed step-size adaptation algorithm

$$\mathbf{d}_{1m}(l+1)=\mathbf{d}_{1m}(l)-\mu\left(\mathcal{P}_{\mathbf{S}_1}^{\perp}\left(\mathbf{r}(l+1)\mathbf{r}(l+1)^{\dagger}-\sigma^2\mathbf{I}\right)\left(\mathbf{S}_1\mathbf{q}_{1m}+\mathcal{P}_{\mathbf{S}_1}^{\perp}\mathbf{d}_{1m}(l)\right)\right) \tag{17}$$

to stochastically estimate $\mathbf{d}_{1m}$. We can replace μ by a step-size sequence μ_l and with certain conditions on μ_l, obtain the Robbins-Monro formulation of the adaptation algorithm. In either case, at iteration $l+1$ we can reconstruct the estimate $\mathbf{c}_{1m}(l+1)$ from $\mathbf{d}_{1m}(l+1)$ by the relation:

$$\mathbf{c}_{1m}(l+1)=\mathbf{S}_1\mathbf{q}_{1m}+\mathcal{P}_{\mathbf{S}_1}^{\perp}\mathbf{d}_{1m}(l+1)\,. \tag{18}$$

Note that the adaptation rule does not require the knowledge of the energy or phase of any of the signals received, and that the convergence here is to the actual decorrelator and not its scaled version. The terms $\mathbf{S}_1\mathbf{q}_{1m}$ and $\mathcal{P}_{\mathbf{S}_1}^{\perp}$ remain the same for each iteration and, therefore, can be pre-computed and stored.

The recursion in (17) followed by (18) can be seen to be equivalent to:

$$\mathbf{d}_{1m}(l+1)=\mathbf{d}_{1m}(l)-\mu\left(\mathbf{r}(l+1)\mathbf{r}(l+1)^{\dagger}-\sigma^2\mathbf{I}\right)\left(\mathbf{S}_1\mathbf{q}_{1m}+\mathcal{P}_{\mathbf{S}_1}^{\perp}\mathbf{d}_{1m}(l)\right)\,, \tag{19}$$

followed by (18). In either adaptation scheme ((17) and (18)) or ((19) and (18)), we observe that all the M signals of user 1 are used explicitly as a group for each decorrelator and we shall refer to this as the Group Signal-space based Blind Adaptive Noncoherent Decorrelator or GS-BAND.

In terms of the decision statistics $z_{1m}(l+1)=\mathbf{c}_{1m}^{\dagger}(l)\mathbf{r}(l+1)$, for $m=1,\cdots,M$, the adaptation algorithm in (19) can be efficiently implemented as

$$\mathbf{d}_{1m}(l+1)=\mathbf{d}_{1m}(l)+\mu\sigma^2\mathbf{c}_{1m}(l)-\mu\mathbf{r}(l+1)z_{1m}^{\dagger}(l+1). \tag{20}$$

The only additional significant computation for the GS-BAND, compared to the SS-BAND, is the multiplication $\mathcal{P}_{\mathbf{S}_1}^{\perp}\mathbf{d}_{1m}(l)$ in (18).

The proof of convergence for the SS-BAND is similar to that for the blind adaptive decorrelator for linear modulation in [5], deviating only due to the difference in the model for the received signal. The analysis, essentially, follows the approach in [7], [8]. Along the same lines, but with some significant changes in the details, we can show convergence for the GS-BAND. The convergence analysis of the fixed step size versions of both the SS-BAND and the GS-BAND in the MSE sense, shows the familiar tradeoff between the value of MSE that we can achieve and the rate of convergence. Further, for the Robbins-Monro formulation of both the adaptive rules, the MSE is shown to go to zero as the number of iterations goes to infinity. The details of the convergence analysis for the GS-BAND can be found in [9].

5 Numerical examples

In this section, we illustrate the performance of the SS-BAND and the GS-BAND for a 3-user, quartenary signalling example for fixed and time-varying step sizes in terms of the Normalized Squared Error (NSE) and the symbol error probability. We define the NSE for user 1 at iteration l as

$$\mathrm{NSE}(l) = \frac{1}{M}\sum_{m=1}^{M} \frac{||\delta_{1m}\mathbf{c}_{1m}^{*} - \mathbf{c}_{1m}(l)||^2}{||\delta_{1m}\mathbf{c}_{1m}^{*}||^2}, \tag{21}$$

where δ_{1m} equals (M/E_{1m}) for the SS-BAND and 1 for the GS-BAND. The NSE is a measure of the average MSE. The twelve signals are generated as normalized random vectors of length or "processing gain" equal to 12. The noise power σ^2 is -20 dB and the user energies for all but the symbol error probability simulations are taken to be $E_{1j} = 10$, $E_{2j} = 10$ and $E_{3j} = 10$ for $j \in \{1, \cdots, 4\}$. They are each equal to 1 for the example on the simulated error probabilities.

Example 1 *Figure 1 compares the peformance of the SS-BAND and GS-BAND w.r.t to NSE. The fixed step-size* μ *is* 0.001 *and the decreasing step-size sequence is chosen as* $\mu_l = \frac{1}{l+8}$*. The starting vector in all cases is chosen as* $\mathbf{c}_{1m}(0) = \mathbf{S}_1\mathbf{q}_{1m}$ *or equivalently* $\mathbf{d}_{1m}(0) = \mathbf{0}$*. Although the NSE is guaranteed to converge to zero for the decreasing step-size versions of both the SS-BAND and GS-BAND, they do so at significantly different rates. Note also that a significant improvement is seen in both the convergence rate and the limiting MSE for the GS-BAND over the SS-BAND for the same fixed step size.*

Example 2 *In Figure 2 we compare the averaged NSE for the GS-BAND for different fixed step sizes, namely,* $\mu = 0.0001, 0.001, 0.01$*. Notice that as the step size becomes smaller, the rate of convergence of the NSE slows down. However, for* $\mu = 0.01$*, the limiting NSE (which is proportional to the step-size) is clearly more than that for the smaller step-size of* $\mu = 0.001$*. Further, it is true for the decreasing step size GS-BAND (as for the SS-BAND) that the initial convergence is fast and becomes slower with increasing number of iterations.*

Example 3 *In this example, we investigate the probability of error performance of the SS-BAND and the GS-BAND by simulation. The signal energies here are chosen as* $E_{1j} = 1$*,* $E_{2j} = 1$ *and* $E_{3j} = 1$ *for* $j \in \{1, \cdots, 4\}$ *and the starting vectors for the adaptive algorithms are the same as in Example 1. Figure 3 plots the NSE and Figure 4 plots the corresponding simulated Symbol Error Rate (SER) curves for the conventional detector, the SS-BAND, the GS-BAND and the deterministic Decorrelator (DEC-GLRT). The post-decorrelative detector used in the latter three cases is the GLRT-based detector. Note that the BER upper and lower bounds for the deterministic decorrelator (see [1]) in Figure 4, which form the bottom-most line in the plot are indistinguishable. For both the adaptive algorithms, the fixed step size is chosen to be* $\mu = 0.001$ *and the decreasing step-size sequence is set to* $\mu_l = 1/(l+8)$*. The significant improvement in SER performance of the GS-BAND over the SS-BAND is seen in both figures in terms of the NSE and the SER.*

6 Conclusions

The problem of blind adaptive noncoherent detection for nonlinear modulation is addressed in this paper. Two blind adaptive noncoherent decorrelators for NMM are obtained that require only the signals of the user of interest, but require no training sequences or knowledge of the user energies. The first one, called SS-BAND, adapts to each of the M decorrelators for any user by employing only the corresponding signal to guide the adaptation algorithm, and can be seen as an extension of the blind adaptive decorrelator for linear modulation presented in [5]. The second adaptive decorrelator uses the entire M-dimensional signal space of each user in the adaptation algorithm, and is called the GS-BAND. Each of the blind adaptive noncoherent decorrelators has a fixed step-size and a decreasing step-size version. Both the fixed step-size and the decreasing step-size (with the step-size sequence satisfying the Robbins-Monro [6] conditions) versions can be shown to converge, in the MSE sense, to the deterministic decorrelator [9]. The former does so with a non-zero, step-size dependent limiting MSE, and the latter with zero limiting MSE. Numerical examples show the superiority of the GS-BAND over the SS-BAND in terms of convergence properties of the MSE and SER. The blind adaptive noncoherent decorrelators followed by the GLRT-based detector form an adaptive detection strategy for NMM that requires only the signals of the user of interest, and is therefore very suitable for distributed implementation.

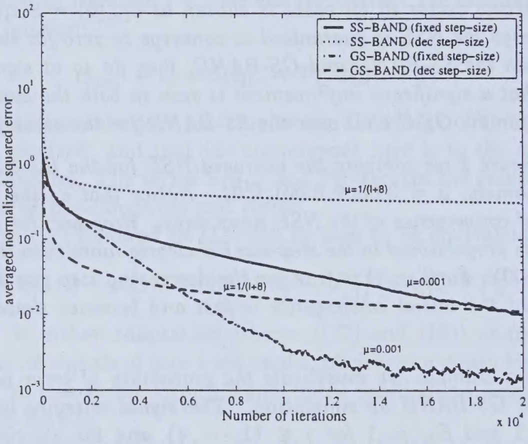


Figure 1: Averaged normalized squared error of the SS-BAND and GS-BAND.

References

[1] M. K. Varanasi and A. Russ, "Noncoherent Decorrelative Multiuser Detection for Nonlinear Nonorthogonal Modulation", *IEEE Transactions of Communications*, vol. 46 , pp. 1675-1684, 1998.

[2] R. Lupas and S. Verdu, "Linear Multiuser Detectors for Synchronous Code-Division Multiple-Access Channels ", *IEEE Transcations on Information Theory*, vol. 35, pp. 123-136, Jan 1989.

[3] U. Madhow and M. L. Honig, "MMSE Interference Suppression for Direct-Sequence Spread-Spectrum CDMA ", *IEEE Transcations on Communications*, vol. 38, pp. 509-519, Apr 1990.

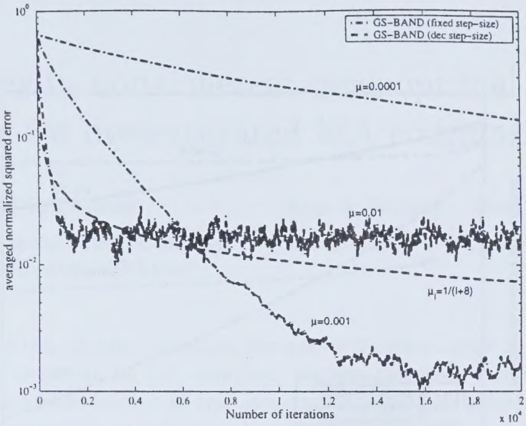


Figure 2: Averaged normalized squared error of the GS-BAND for different step sizes.

[4] M. L. Honig, U. Madhow and S. Verdu, "Blind Adaptive Multiuser Detection", *IEEE Transcations on Information Theory*, vol. 41, pp. 994-960, Jul 1995.

[5] S. Ulukus and R. D. Yates, "A Blind Adaptive Decorrelating Detector for CDMA Systems", *IEEE Journal on Selected Areas of Communications*, Vol. 16, No. 8, pp. 1530-1541, Oct 1998.

[6] H. Robbins and S. Monro, "A Stochastic Approximation Method", *Annals of Mathematical Statistics*, Vol. 22, pp. 400-407, 1951.

[7] D. J. Sakrison, "Stochastic Approximation: A recursive method for solving regression problems", *Advances in Communication Systems 2*, A. V. Balakrishnan, Ed. New York: Academic, pp. 51-106, 1966.

[8] Ya. Z. Tsypkin, *Adaptation and Learning in Automatic Systems*, Academic Press, 1971.

[9] D. Das and M. K. Varanasi, "Blind Adaptive Noncoherent Multiuser Detection for Nonlinear Modulation", submitted to the *IEEE Transactions of Communications* for publication, Aug 1999.

[10] R. Horn and C. Johnson, *Matrix Analysis*, Cambridge University Press, 1993.

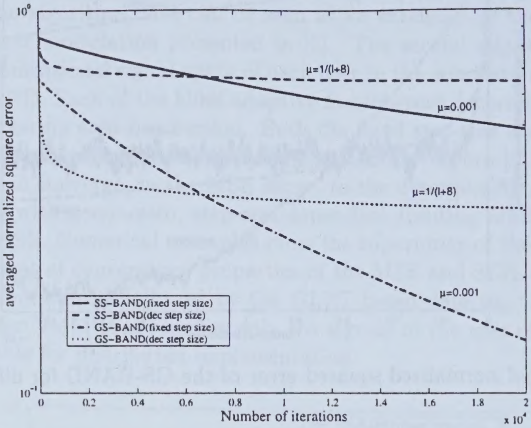


Figure 3: Averaged normalized squared error of the SS-BAND and GS-BAND.

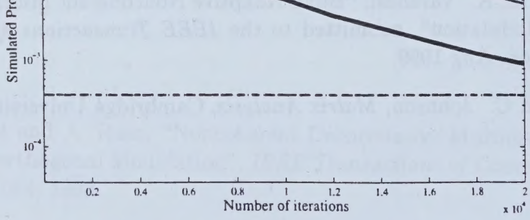


Figure 4: Simulated symbol error probability curves correponding to Figure 3 for the conventional detector, SS-BAND, GS-BAND and deterministic decorrelative-GLRT (DEC-GLRT) detector upper and lower bounds.

Low complexity non-coherent near-optimal multiuser detection for oversaturated MA communication

Rachel E. Learned
Sanders, A Lockheed Martin Company
Rachel.E.Learned@LMCO.com

Alan S. Willsky Don M. Boroson
Massachusetts Institute of Technology

The general problem of joint detection for non-orthogonal users in an MA communications system is known to be N-P complex. As was shown in a recent paper by the authors that with the introduction of tree structured cross-correlations among user signature waveforms, optimal joint detection is possible with a very low order polynomial complexity procedure. Moreover, in a separate work by Ross and Taylor, a tree-structured signature set construction was developed to squeeze additional users into an ideal orthogonal MA system causing negligible performance loss relative to the orthogonal system when optimal symbol by symbol joint detection is used. The use of the tree joint detector with the Ross/Taylor signature sets allows, in principle, for over-saturation (more users than signal space dimensions) with comparable performance and computational complexity to the corresponding orthogonal MA system supporting many fewer users. A common assumption for the development of decision feedback and onion peeling types of multiuser detectors is phase coherence in which all user phases are known at the receiver; prior work of the authors has been restricted to this unrealistic coherent case. This paper presents a near-optimal low complexity phase acquisition algorithm for each new user as it enters the tree-structured MA system. As was the case for the optimal coherent tree-detector, the complexity of the non-coherent near-optimal multiuser tree-detector is low order polynomial in the number of users for typical tree structures of interest. Moreover, using only a short training sequence during acquisition, the performance of the non-coherent tree-detector in which user phases are initially unknown at the receiver nearly matches that of the optimal coherent tree-detector in which user phases are known at the receiver. Empirical analysis shows the non-coherent tree-detector to be robust to typical estimate errors on each user's phase.[1]

1 Introduction

Adopting the commonly-used vector space framework ([1]), the multiuser joint detection problem may be stated as follows: for a given set of K user waveforms represented in signal space by the set of signal vectors, $\{\mathbf{s}_k\}_1^K$, $\mathbf{s}_k \in \mathbb{C}^N$, the general uncoded detection problem is to compute an estimate of weights, $\mathbf{b}$, from an observation $\mathbf{r} \in \mathbb{C}^N$ for which

$$\mathbf{r} = \sum_{k=1}^{K} b_k \mathbf{s}_k e^{j\phi_k} + \sigma \mathbf{n} = \mathbf{S}\Phi\mathbf{b} + \sigma\mathbf{n}. \qquad (1)$$

---

[1] This research is supported in part by the Department of the Air Force under contract F19628-95-C-002 and in part by the Air Force Office of Scientific Research under Grant F49620-95-1-0083 and the Army Research Office under Grant DAAL03-92-G-0115.

Time-bandwidth restrictions on any communication system impose a limit in the dimension, N, of the space of possible user waveforms. The N-dimensional signal space, then, corresponds to $\mathbb{C}^N$, the N-dimensional space of complex vectors. The following notation is also used:

- K is the number of users.
- $\mathbf{b} \in \Gamma = \{[b_1 \cdots b_K]^T \mid b_i \in P_i\}$, where P_i is some finite set of amplitudes (real or complex) and b_i is independent and identically distributed (i.i.d.), equally likely. For P_i having M elements, this is M-ary signaling.
- ϕ_k is user k's phase and is arbitrary, i.e. neither the user nor the receiver has control over the value of the user phases as seen at the receiver.
- $\mathbf{S} = [\mathbf{s}_1, \ldots, \mathbf{s}_K]$ is an $N \times K$ matrix whose columns are user signal vectors as seen at the receiver. The oversaturated case is when there are more users than signal dimensions, i.e. $K > N$.
- $\Phi = diag\{e^{\phi_k}\}_1^K$
- $\mathbf{n}$ is a vector of independent, circularly Gaussian, zero mean, unit variance complex random variables.
- σ is the noise standard deviation.

Equation (1) is a reasonable model for many MA systems. The geosynchronous satellite uplink channel, for example, is well modeled by Equation (1) where the received waveforms are, simply, translated and attenuated replicas of the transmitted waveforms.

The optimal (Maximum Likelihood) coherent joint detector for Equation (1), given that all user signatures, including the phases, are correctly known at the receiver, is ([2])

$$\hat{\mathbf{b}} = \arg \min_{\mathbf{b} \in \Gamma} ||\mathbf{r} - \mathbf{S}\Phi\mathbf{b}||^2. \tag{2}$$

Notice that the above coherent detector requires a search over all possible realizations of the K-element vector $\mathbf{b}$. If each user employs M-ary signaling, the number of possible realizations of $\mathbf{b}$ is M^K. For typical values of M and K, this complexity is impractical. In [3] we showed that when a hierarchical cross-correlation structure is imposed on the user waveforms, optimal detection can be achieved with a coherent tree-structured receiver having complexity that is, in typical cases, a *low-order-polynomial* in the number of users.

Many joint detection schemes, including the optimal joint detector and the optimal low complexity tree-detector, require the knowledge of the received signal waveforms, $\mathbf{S}$ and Φ. The set of possible received signals is constructed by the optimal detector using replicas of the received signature signals (including phases) that are stored at the receiver. If we have incorrect knowledge or no knowledge of the signal phases, our recreation of the set of possible received signals will be incorrect and the optimal procedure will not work. This is also the case for a host of popular suboptimal multiuser detectors in the literature ([4, 5], for example).

This paper continues our investigation of a practical low complexity optimal multiuser detection scheme for the over-saturated case in which the set of user waveforms is linearly dependent ($K > N$). Recent work by Sankaran and Ephremides ([6]) finds a low complexity optimal detector for a set of linearly independent signals having non-positive cross correlations. This low complexity optimal scheme as well as other suboptimal methods proposed in the literature (such as the non-optimal joint detectors described in the text by Verdu[7]) minimally require $K \leq N$ and, thus, are not intended for extension to the over-saturated scenario. Conversly, the work described in this paper is based on a new foundation for MA communications for which the primary goal is to increase the number of simultaneous users such that $K > N$ [3],[8].

Our previous work on low complexity optimal joint detection focussed on the coherent case for which user phases were correctly known at the receiver [3]. In an actual coherent system a user's phase is estimated by the receiver during the first several symbols transmitted by the user. The estimated phase is taken to be true for subsequent symbols or is tracked appropriately.

It is important, then, for a coherent multiple access joint detection scheme to incorporate a low complexity phase estimation first stage. In this paper the estimation of a user's phase is incorporated into the tree detection procedure; a non-coherent low complexity multiuser tree-structured detector is proposed and an empirical performance analysis is offered. Using only a short training sequence during acquisition, the performance of the non-coherent tree-detector in which user phases are unknown at the receiver was found to nearly match that of the optimal coherent tree-detector in which user phases are known at the receiver. The complexity of the non-coherent near-optimal multiuser tree-detector is low order polynomial in the number of users for typical tree-structures of interest in practical scenarios.

2 Background

2.1 Tree Structure

Tree structure among the cross correlations of user signatures was proposed in [3] to combat complexity of the optimal receiver. Conceptually, the columns of the matrix $\mathbf{S}$, can be assigned to the nodes of a tree. For a specific example, see Figure 1. The tree pictorially conveys the following required relationships among user signal vectors.

- Each vector across a given level of the tree is orthogonal to all other vectors at that level.
- A signal vector is correlated only with its ancestor vectors and its descendant vectors.

The tree need not be of uniform construction as shown in the example of Figure 1. In general, the complexity of the optimal detector is exponentially proportional to the total number of levels in the tree, which, for the majority of useful trees, is far less than the number of users[3].

The maximum a posteriori tree detection algorithm is derived and detailed in [3]. The optimal algorithm takes advantage of the tree structure, sweeping through the tree from bottom to top, creating a table of conditional weight estimates at each node. The table of decisions at a given node is conditioned upon weight decisions of the ancestors and is a function of weight decisions of the descendants. Since each conditional estimate table requires an entry for each possible combination of weights at all ancestor nodes, the number of computations needed to create a table is exponential in the number of ancestors. As the procedure moves from the bottom to the top of the tree the number of decisions made at each level decreases exponentially until there is only one decision associated with the top node of the tree. The full weight vector estimate for all user weights is a by-product of the last decision at the top of the tree. The total complexity is extremely low; for most trees, with the exception of some erratic special cases, the complexity per bit is a low order polynomial in the number of users[3]. For a quad tree cross correlation structure such as that shown in Figure 1, the complexity is $O(K^{\frac{3}{2}})$.

2.2 Phase Accuracy Requirement

In an MA system, a channel is set typically set aside for initital interaction between the base station and the users. Specifically, if a new user wishes to obtain a place in the system, it makes a request to the base station over this separate acquisition channel. If the system can accommodate this new user, the base station assigns this user a signature waveform (which might consist of a signature envelope, modulation frequency, and time slot). During the acquisition process, the base station refines any timing or power demands it might have on this new user by sending control signals telling the new user to make the necessary adjustments.[1] Although the phase of the new user's transmission can be estimated during acquisition, this phase will not remain the same once the user begins transmitting at it's assigned modulation frequency. Since coherent changeover from the acquisition channel to the transmission channel does not yet exits, the phase of this new user will be unknown to the receiver once communication commences and it will need to be acquired through some type of estimation procedure.

The performance of the ML joint detector depends upon the Euclidean distances among point in the noise-free received constellation (compirsed of all possible combinations of data modulated user signature pulses). The positions of points in this constellation is dependent upon the relative powers and cross correlations of the user signatures and would be different for every signature set. Througout this paper we gain insight through empirical analysis via simulations using a specific type of signature set that is amenable for the geo-synchronous satellite scenario, specifically, the over-saturated 21 user, unit energy minimum distance set of Ross and Taylor ([9]) for which

$$\mathbf{S} = \begin{bmatrix} & 1/2 & 0 & 0 & 0 & 1/4 \\ & 1/2 & 0 & 0 & 0 & 1/4 \\ & 1/2 & 0 & 0 & 0 & 1/4 \\ & 1/2 & 0 & 0 & 0 & 1/4 \\ & 0 & 1/2 & 0 & 0 & 1/4 \\ & 0 & 1/2 & 0 & 0 & 1/4 \\ & 0 & 1/2 & 0 & 0 & 1/4 \\ \mathbf{I}_{16} & 0 & 1/2 & 0 & 0 & 1/4 \\ & 0 & 0 & 1/2 & 0 & 1/4 \\ & 0 & 0 & 1/2 & 0 & 1/4 \\ & 0 & 0 & 1/2 & 0 & 1/4 \\ & 0 & 0 & 1/2 & 0 & 1/4 \\ & 0 & 0 & 0 & 1/2 & 1/4 \\ & 0 & 0 & 0 & 1/2 & 1/4 \\ & 0 & 0 & 0 & 1/2 & 1/4 \\ & 0 & 0 & 0 & 1/2 & 1/4 \end{bmatrix}, \tag{3}$$

---

[1] This procedure is typical of state of the art military satellite communications in which the base station is the satellite.

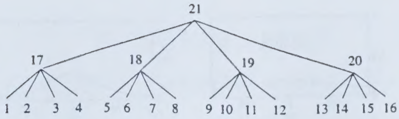


Figure 1: Correlation tree for a unit energy minimum distance signature set of 21 users in 16 dimensions.

where I_{16} is the 16 dimensional identity matrix. The tree corresponding to the cross-correlations among signature waveforms in this minimum distance set is shown in Figure 1.

We tested the tree joint detector (TJD) against phase mismatches in all users; a wrapped Gaussain distribution having zero mean and standard deviation, $\sigma_m = 4°$, produced only a 1% loss in E_b/N_o relative to the case in which all users' phases were perfectly known.[2]. For the remaining development in this paper, we accept a standard deviation $\sigma_m \leq 4°$ in phase error as tolerable.

3 Partially coherent one-shot phase estimation

As is shown in this section, a one-shot phase estimation algorithm, applied during a single bit or symbol duration, is capable of achieving accuracies of 24° for typical SNRs. This error is much higher than our $\sigma_m = 4°$ requiremet. The oneshot phase estimate will serve as the foundation for the estimation procedure that makes use of a sequence of symbol transmissions to achieve the needed phase estimate accuracy.

The optimal one-shot estimator is derived first for the general case of the receiver having partial to no information about the true phase to be estimated. The no-information scenario corresponds to the requirement for phase acquisition upon entry of a user into the tree-structured multiuser system. The partial-information scenario is treated so that we may have the option of refining or improving a phase estimate as desired. No special structure among user signature waveforms is assumed for the derivation. Tree structure among user signatures is then introduced to facilitate reduction in computational complexity.

3.1 The Estimator

The maximum a posteriori (MAP) joint weight/phase estimator is derived for the case in which all but one user's phase is known at the receiver. This one-at-a-time approach to phase estimation is valid since, in an actual communication system, new users join the system one at a time.

The received signal is modeled by the complex vector equation which explicitly shows the uncertain phase, ϕ_K, to be estimated,

$$\mathbf{r} = \mathbf{S}\Phi\mathbf{b} + b_K \mathbf{s}_K e^{j\phi_K} + \sigma\mathbf{n}. \tag{4}$$

$\mathbf{S}$ now has as its columns the $K-1$ user signature vectors that correspond to the users with known phases. The known phases are collected in $\Phi = diag[\{e^{j\phi_k}\}_1^{K-1}]$. The (K-1)-element information weight vector, $\mathbf{b}$, has a finite number of possible realizations enumerated in the discrete set Γ.

---

[2] For more details see [10]

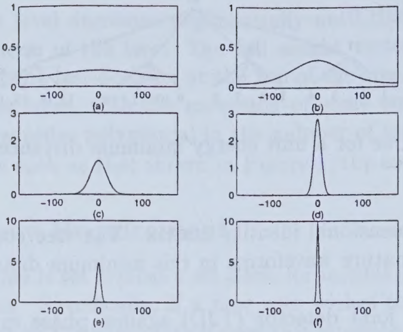


Figure 2: The exponential cosine PDF is assumed to be the distribution for ϕ_K, where $p_{\phi_K}(\phi) = \frac{e^{\alpha\cos\phi}}{2\pi I_o(\alpha)}$, $-180^o \leq \phi_K < 180^o$. This PDF is shown for six different values of α. (a) $\alpha = 0.25$. Note that as α approaches zero the PDF approaches a uniform distribution. (b) $\alpha = 1$ (c) $\alpha = 10$ (d) $\alpha = 50$ (e) $\alpha = 150$ (f) $\alpha = 600$ Note that as α approaches infinity the PDF approaches a Dirac.

Notice that Equation (4) allows for the uncertain-phase user to transmit a weight, b_K. If the phase for user K is completely unknown at the receiver, it is best if user K transmits a known training weight. The use of training sequences during phase acquisition is common practice in today's systems. If, on the other hand, user K's phase was partially known to be within some range of moderate arc length, user K could transmit an information weight (i.e., a weight not known to the receiver a priori) without hindering the receiver's ability to estimate user K's phase.

The optimal one-shot joint weight-phase estimator is given for the general case for which user K transmits an information weight; this would be the case during phase refinement. Without loss of generality, we may think of user K's presumed phase as being zero radians and the random error in our knowledge of this phase as ϕ_K. A reasonable assumption for the prior distribution on the phase error is

$$p_{\phi_K}(\phi) = \frac{e^{\alpha\cos\phi}}{2\pi I_o(\alpha)}, \tag{5}$$

where $I_o(\alpha)$ is the modified Bessel function of zeroth order. This exponential cosine PDF has been shown by Viterbi ([11]) to be the PDF for the MAP phase estimate of the phase-locked loop applied to a single user in white Gaussian noise. Figure 2 shows this PDF for several values of α. The parameter α allows the phase to range from a uniform random variable when $\alpha = 0$, to a deterministic variable, when $\alpha = \infty$.

The MAP joint weight/phase estimate, derived in [10], may be found in two steps. First, find the weight estimates

$$\{\hat{\mathbf{b}}, \hat{b}_K\} = \arg\min_{\{\mathbf{b}, b_K\}} [\, \frac{F_K(\mathbf{b}|\mathbf{r})}{\sigma^2} + \frac{\varepsilon_K(b_K)}{\sigma^2} - |\frac{2}{\sigma^2}\mathcal{X}(\mathbf{b}, b_K|\mathbf{r}) + \alpha| \,] \tag{6}$$

then find user K's phase estimate

$$\hat{\phi}_K = \tan^{-1}\frac{\Im\{\mathcal{X}(\hat{\mathbf{b}}, \hat{b}_K|\mathbf{r})\}}{\Re\{\mathcal{X}(\hat{\mathbf{b}}, \hat{b}_K|\mathbf{r})\} + \alpha\frac{\sigma^2}{2}}, \tag{7}$$

where

$$F_K(\mathbf{b}|\mathbf{r}) = ||\mathbf{r} - \mathbf{S}\Phi\mathbf{b}||^2, \tag{8}$$

$$\varepsilon_K(b_K) = ||b_K \mathbf{s}_K e^{j\phi}||^2 = |b_k|\,||\mathbf{s}_K||^2, \tag{9}$$

and

$$\mathcal{X}(\mathbf{b}, b_K|\mathbf{r}) = b_K^* \mathbf{s}_K^T(\mathbf{r} - \mathbf{S}\Phi\mathbf{b}). \tag{10}$$

For phase acquisition, the receiver has no prior knowledge of user K's phase. User K is required to transmit a known training weight. To reflect this scenario, we set $\alpha = 0$ and $\hat{b}_K = b_K$ in the estimator of Equations (6) and (7).

3.2 A reduction in complexity for tree structured signature sets

In general, the optimal joint weight/phase estimator given in the previous section requires trying every possible realization of $\mathbf{b}$, thus $|\Gamma|$ comparisons are done, where $|\Gamma|$, the number of elements in the set Γ, is exponential in the number of users. If the user signature vectors are tree-structured, a vast savings in computational complexity of the joint estimator may be realized. $F_K(\mathbf{b}, b_K|\mathbf{r})$ of Equation (8) is separable when conditioned on the weights that correspond to the values of b_i for node i an ancestor node of node K ([3]). The other term in Equation (6), $|\mathcal{X}(\mathbf{b}, b_K|\mathbf{r})|$, is not separable by conditioning, but may be significantly reduced, since

$$\mathbf{s}_K^T \mathbf{s}_n = 0 \quad \forall n \notin adK = dK \cup aK,$$

where adK denotes the set of indices that correspond to users that are either ancestors or descendants of user K. Similarly, dK (aK) corresponds to the descendant (ancestor) nodes to node K on the cross-correlation tree for the signature set of interest.

It follows that as a by-product of the tree structure, $\mathcal{X}(\mathbf{b}, b_K|\mathbf{r})$ is a function of the weights associated with user K's ancestors and descendants and is *independent* of the rest of the weights corresponding to user signatures that are orthogonal to user K's signature. For tree-structured signature sets,

$$\mathcal{X}(\mathbf{b}, b_K|\mathbf{r}) = \mathcal{X}(\mathbf{b}_{adK}, b_K|\mathbf{r}) = b_K^* \mathbf{s}_K^T(\mathbf{r} - \mathbf{S}_{adK}\Phi_{adK}\mathbf{b}_{adK}), \tag{11}$$

where $\mathbf{S}_{adK}$ has as its columns the signature vectors of the users corresponding to the ancestor and descendant nodes to node K on the cross-correlation tree for the given signature set. Likewise, Φ_{adK} and $\mathbf{b}_{adK}$ hold the phases and weights, respectively, of the same ancestor and descendant users.

$F_K(\mathbf{b}|\mathbf{r})$ can be decomposed into N conditionally independent terms, corresponding to the N users at the bottom of the tree ([3]). This separation allows for a tree climbing procedure that creates conditional weight estimate tables at each node. Moreover, with one unknown-phase user, the minimization in Equation (6) can also be carried out through a tree climbing procedure of table creations.

For this unknown-phase case there are four types of tables. Figure 3 pictorially shows the four different types of tables and where they occur on the tree relative to the node at which the unknown-phase user is located. For the example in Figure 3, user 12 is the last addition to the system. The phase of user 12 is not known, but the phases of the other users are assumed to have been accurately acquired prior to the addition of user 12. Below, a description of each type of table is given. A detailed example showing the construction of each table is offered in [10].

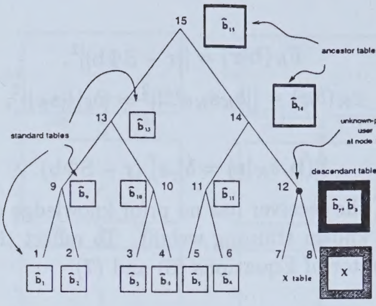


Figure 3: Example of the non-coherent weight/phase estimator and the tables it constructs.

Standard table: Since $\mathcal{X}(\mathbf{b}_{adK}|\mathbf{r})$ is independent of $\mathbf{b}_{\overline{adK}}$ (the weights at nodes n, $\forall n \notin adK$), a **standard table** of conditional weight estimates may be constructed at all nodes n, $\forall n \notin adK$. The estimate that is stored in the standard table at node n for a specific realization of $\mathbf{b}_{an}$, the ancestor weights, is found from

$$\hat{b}_n(\mathbf{r}|\mathbf{b}_{an}) = \arg \min_{b_n} ||\mathbf{r} - \mathbf{s}_n b_n e^{j\phi_n} - \mathbf{S}_{an}\Phi_{an}\mathbf{b}_{an} - \mathbf{S}_{dn}\Phi_{dn}\hat{\mathbf{b}}_{dn}(\mathbf{r}|b_n, \mathbf{b}_{an})||^2,$$

where $\hat{\mathbf{b}}_{dn}(\mathbf{r}|b_n, \mathbf{b}_{an})$ is the collection of weight estimates for the descendant users of user n for a given set of weights for the ancestors of user n. The weight estimates, $\hat{\mathbf{b}}_{dn}(\mathbf{r}|b_n, \mathbf{b}_{an})$, can easily be obtained from a succession of simple standard table look-ups at the descendent nodes of node n.

$\mathcal{X}$-table: Due to the existence of the unknown phase at node K of the tree, the weight estimates corresponding to nodes dK cannot be decoupled through conditioning. In other words, since there are infinitely many possibilities for the phase at node K, weight estimates at nodes dK cannot be conditioned upon ϕ_K. Instead, a single, large table, the **$\mathcal{X}$-table**, may be constructed. For each possible realization of $\mathbf{b}_{adK}$, this table has one entry, namely the complex scalar resulting from the calculation of Equation (11).

Descendant table: The next table that can be constructed is a table of joint conditional weight estimates for the descendants of node K. This table is most conveniently placed at node K (node 12 in the example of Figure 3). The joint estimate that would be entered into the **descendant table**, for a specific realization of $\mathbf{b}_{aK}$, is found from

$$\hat{\mathbf{b}}_{dK}(\mathbf{r}|\mathbf{b}_{aK}) = \arg \min_{\mathbf{b}_{dK}} ||\mathbf{r} - \mathbf{S}_{aK}\Phi_{aK}\mathbf{b}_{aK} - \mathbf{S}_{dK}\Phi_{dK}\mathbf{b}_{dK}||^2 - 2|\mathcal{X}(\mathbf{b}_{aK}, \mathbf{b}_{dK}|\mathbf{r})|.$$

Note that the values of the conditional estimates of $\mathbf{b}_{\overline{adK}}$ are independent of $\mathbf{b}_{dK}$ and, thus, do not appear in the minimization equation for the creation of the descendant table. This is due to the conditionally separable property of $F_K(\mathbf{b}|\mathbf{r})$ that was shown in [3].

Ancestor tables: The estimation algorithm is completed by successively creating tables for the remainder of the tree, at the nodes aK, starting first, with node pK, the parent node to node K, and ending with the root node. At a node $i \in aK$, an ancestor table is constructed by calculating a conditional estimate for every possible realization of $\mathbf{b}_{ai}$ from

$$\hat{\mathbf{b}}_i(\mathbf{r}|\mathbf{b}_{ai}) = \arg \min_{b_i} \mathcal{F}_i(b_i, \mathbf{b}_{ai}|\mathbf{r}) - 2|\mathcal{X}(\hat{\mathbf{b}}_{dK}(\mathbf{r}|\hat{\mathbf{b}}_{\vartheta}(\mathbf{r}|b_i\mathbf{b}_{ai})), \hat{\mathbf{b}}_{\vartheta}(\mathbf{r}|b_i\mathbf{b}_{ai}), b_i, \mathbf{b}_{ai}|\mathbf{r})|$$

Table 1: Computational complexity of tree algorithm for binary signaling ($M = 2$) and a quad tree as a function of L_n. L is the total number of levels in the tree, K is the total number of users (nodes on the tree), and L_n is the level at which the unknown-phase user sits on the tree.

L	K	$L_n = L$	$L_n = L - 1$	$L_n = L - 2$	$L_n = L - 3$
3	21	73	117	2.0972e+06	
4	85	585	673	4.1947e+06	3.8686e+25
5	341	4681	4857	8.3930e+06	7.7371e+25

where

$$\begin{aligned}\mathcal{F}_i(b_i, \mathbf{b}_{ai}|\mathbf{r}) \;=\; || \;\; & \mathbf{r} - \mathbf{s}_i e^{j\phi_i} b_i - \mathbf{S}_{ai}\Phi_{ai}\mathbf{b}_{ai} \\ & - \mathbf{S}_{dK}\Phi_{dK}\hat{\mathbf{b}}_{dK}(\mathbf{r}|\hat{\mathbf{b}}_{\vartheta}(\mathbf{r}|b_i\mathbf{b}_{ai}), b_i, \mathbf{b}_{ai}) \\ & - \mathbf{S}_{\varsigma}\Phi_{\varsigma}\hat{\mathbf{b}}_{\varsigma}(\mathbf{r}|b_i\mathbf{b}_{ai}) \;\; ||^2\end{aligned}$$

where $\vartheta = aK\backslash\{i, ai\}$, i.e., the set of ancestor nodes to node K that are not in the set of nodes $\{i, ai\}$. Note that $\hat{\mathbf{b}}_{\vartheta}(\mathbf{r}|b_i\mathbf{b}_{ai})$ can be retrieved from successive table look-ups of ancestor tables previously constructed at the nodes below node i and above node K. Recall that $\hat{\mathbf{b}}_{dK}(\mathbf{r}|\hat{\mathbf{b}}_{\vartheta}(\mathbf{r}|b_i\mathbf{b}_{ai}))$ can be obtained by a simple look-up in the descendant table at node K. In the last term of $\mathcal{F}_i(b_i, \mathbf{b}_{ai}|\mathbf{r})$, $\varsigma = di\backslash dK$, i.e., the set of descendant nodes of node i that are not descendants of node K. The conditional weight estimates for these nodes are collected together in $\hat{\mathbf{b}}_{\varsigma}(\mathbf{r}|b_i\mathbf{b}_{ai})$ and may be obtained from simple successive look-ups of either the standard tables or just-constructed ancestor tables.

3.3 Computational Complexity

The complexity of the one-shot joint weight/phase tree estimation procedure (JWPTE) is counted as the number of comparisons required to construct the standard tables, the ancestor tables, the descendant table and the number of rows in the $\mathcal{X}$-table. The complexity has been calculated in Learned's thesis ([12]) for a uniform tree-structured MA system in which all users employ M-ary signaling and is omitted for brevity. The most complex portions of the detection procedure relate to the descendant table and the $\mathcal{X}$-table for which the complexity is $O(M^{|adn|})$ where $|adn|$ is the number of ancestors and descendants to user n, the unknown-phase user. The complexity can be quite high if the unknown-phase user is high up in the tree since this is the condition that gives the largest value for $|adn|$. Table 1 shows the complexity for three different quad-tree-structured BPSK systems having a total number of levels, L, that vary from three to five levels and for the level of the unknown-phase user, L_n, that varies from bottom to top.

To overcome the high complexity when the unknown-phase user is high up in the tree, a near-optimal version of the joint weight/phase estimation procedure can bring the complexity back down to a low order polynomial in the number of users. This near-optimal approach, proposed and discussed in [12], assumes that the unknown phase is from a finite set of relatively finely discretized phases.

4 Multi-symbol Phase Estimation

The phase estimate obtained with the one-shot JWPTE for a single unknown phase user in the presence of a 4° mismatch on all other users was emperically found to be 24° for our example corresponding to a cross correlation structure in Figure 1 for which the phase of user 17 was unkown. This is several times the needed accuracy of 4°. Clearly, we must estimate the new user's phase over several symbols to achieve the required accuracy.

The optimal multi-symbol phase estimator for uncoded transmission in a tree structured multiple access system is derived in [12]. For all practical cases the complexity of the optimal multi-symbol JWPTE is prohibitive.

Two low complexity estimation procedures are proposed in this section. These sub-optimal low complexity procedures use multiple consecutive symbols of data to achieve the necessary accuracy in the phase estimate of the new or unknown-phase user.

4.1 The Simple Average Phase Estimate

First we propose a simple average of multiple consecutive one-shot phase estimates. A block diagram of this procedure is shown in Figure 4-a. When a new user enters the system we re-set the symbol counter to zero, $i = 0$ and the detector switches from the standard mode to the non-coherent JWPTE mode. The new user must send a pre-agreed upon training sequence for its first T weight transmissions. The detector calculates a running average of the new user's phase estimates calculated by the JWPTE while passing along the weight estimates for all the other users to the detector output. Upon reception of the T^{th} symbol, the the running average of the new user's phase estimate at time symbol $i = T - 1$ is passed to the TJD to be used as the true phase and the detector switches back to the coherent TJD mode. The new user's phase estimate is

$$\hat{\phi}_k = \sum_{i=0}^{T-1} \hat{\phi}_k[i]/T. \tag{12}$$

Also at symbol T, the new user begins to transmit data.

Since each of the phase estimates created by the JWPTE is i.i.d. with a standard deviation denoted by σ_e, we have

$$\sigma_{SA}^2 = \sigma_e^2/T, \tag{13}$$

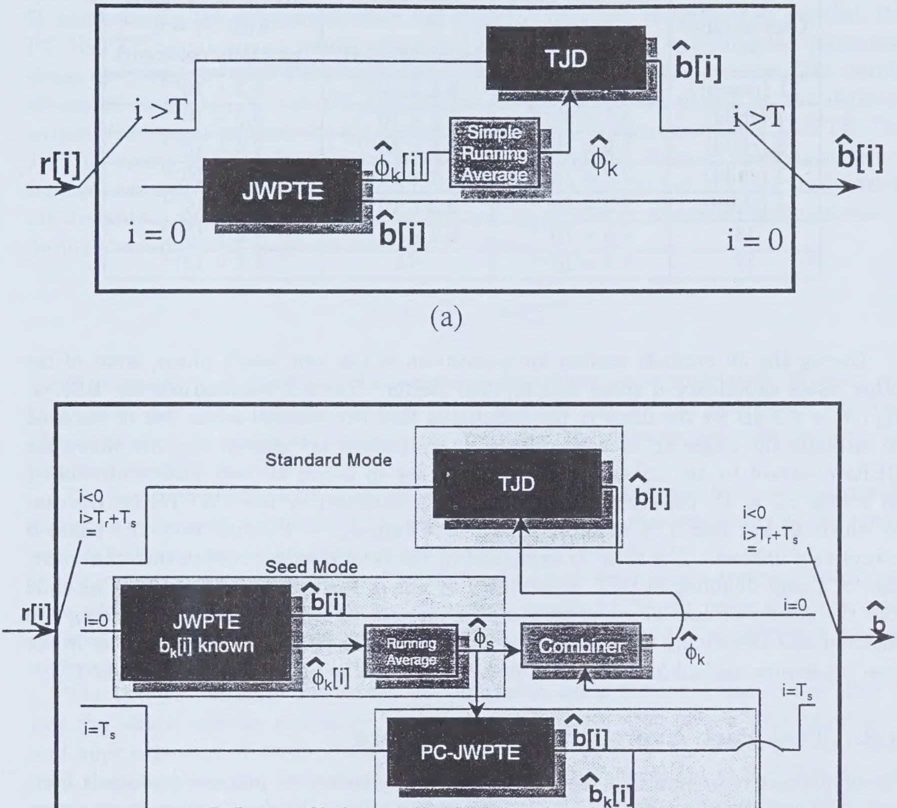
where we denote the simple average phase estimate standard deviation by σ_{SA}.

If we require a 4° standard deviation of the simple average phase estimate for our simulation example that found $\sigma_e = 24°$ at $E_b/N_o = 6.2$ dB, we must set

$$T = \sigma_e^2/\sigma_{SA}^2 = (24°/4°)^2 = 36. \tag{14}$$

The complexity of the phase estimation procedure per bit is the same as the complexity of the JWPTE discussed in Section 3.3.

Simulation of the simple average phase estimation procedure for the 21-user unit-energy signature set in which user 17 has the unknown phase and all other users' phases are known to within a 4° standard deviation was performed. The simulation for $E_b/N_o =$ 6.2 dB showed that the desired accuracy of a 4° standard deviation in the phase estimate for user 17 is possible with 39 training symbols. During the training time, no new users are allowed to enter the system.


TJD
$\hat{b}[i]$
i >T
r[i]
Simple Running Average
$\hat{\phi}_k[i]$
$\hat{\phi}_k$
JWPTE
i = 0
Standard Mode
i<0
$i \geq T_r+T_s$
Seed Mode
JWPTE $b_k[i]$ known
Running Average
$\hat{\phi}_s$
Combiner
i=0
$\hat{b}$
$i=T_s$
PC-JWPTE
$\hat{b}_k[i]$
Refinement Mode
JWPTE $b_k[i] = \hat{b}_k[i]$
$\hat{\phi}_a$

(a)

(b)

Figure 4:

Table 2: BER for the affected users in the 21 user unit energy minimum distance tree-structured BPSK MA system due to an unknown phase for user 17. $E_B/N_o = 6.2$ dB and the phase mismatch on the other users has a standard deviation of $\sigma_m = 4°$. Note that these BER estimates are within 9-13 % of the true values with a probability of 0.95.

User number	Standard mode (coherent)	Seed mode (non-coherent)	Refinement mode with $T_s = 6$ (partially coherent)
21 (parent)	4.9×10^{-3}	8.1×10^{-3}	5.2×10^{-3}
1 (child)	3.2×10^{-3}	5.2×10^{-3}	3.3×10^{-3}
2 (child)	2.6×10^{-3}	5.1×10^{-3}	2.6×10^{-3}
3 (child)	3.7×10^{-3}	4.8×10^{-3}	3.5×10^{-3}
4 (child)	2.5×10^{-3}	3.3×10^{-3}	2.2×10^{-3}
18	4.5×10^{-3}	5.9×10^{-3}	4.8×10^{-3}
17	5.3×10^{-3}	NA	5.3×10^{-3}

During the 39 symbols needed for acquisition of the new user's phase, some of the other users experience a small loss in their BERs. Table 2 summarizes the BER at $E_B/N_o = 6.2$ dB for the users in the simulation that experienced a loss due to the need to estimate the phase on user 17. Three BER columns are shown, the first shows the BERs obtained by the coherent TJD for the case in which all user phases are known to within $\sigma_m = 4°$, the second shows the BERs obtained by the JWPTE for the case in which all but user 17's phase is known to within $\sigma_m = 4°$ while user 17's phase is completely unkown. The third is explained in the next section. Notice that the worst-case of a near doubling in BER is seen only by one of the children of user 17. One child and the parent of user 17 experience a 65% increase in BER while the remaining two children and the sibling experience only a 30% increase in BER. All other users in the tree experience negligible or no BER increases due to the unknown phase of user 17.

4.2 Two-Part Average Phase Estimate

As an alternative to sending a 39-symbol training sequence, we propose a two-part average. The complexity is only slightly greater than that for the simple average procedure described in the previous section; the benefit is twofold. First, the new user sends only a very brief training sequence and can begin to send actual data sooner than was possible in the simple averaging procedure of the previous section. Second, the long exposure of affected users to the higher bit error rates produced by the simple averaging procedure (resulting from use of the non-coherent JWPTE) is greatly shortened.

A block diagram of the two-part phase estimation procedure is shown in Figure 4-b. When a new user enters the system. We reset the symbol counter to $i = 0$ and switch from the standard mode to the **seed mode** which uses the JWPTE. The new user must send a training sequence for its first T_s weight transmissions. While in the seed mode, the detector will calculate a **seed estimate** for the new user's phase. This seed estimate will have a higher standard deviation than that which is ultimately required by the coherent TJD but good enough for reliable operation of the partially coherent joint weight phase estimator (PC-JWPTE), derived in Section 3.[3] While in the seed mode, the detector calculates a running average of the new user's phase estimates calculated by the JWPTE and passes along the weight estimates for all the other users to the detector output. At the start of symbol T_s, the detector passes the seed phase estimate, $\hat{\phi}_s = \sum_{i=0}^{T_s-1} \hat{\phi}_k[i]/T_s$, to the PC-JWPTE and switches into the **refinement mode** for T_r symbols. In the refinement mode the detector uses two estimators in parallel, the PC-JWPTE to calculate a weight estimate for each of the users, including the uncertain-phase user, and the JWPTE to calculate a phase estimate of the new user. The weight estimates calculated by the PC-JWPTE for all the users are passed to the detector output; the weight estimate for the uncertain phase user is also passed to a JWPTE. The JWPTE treats this weight estimate stream as if it were the "known" training sequence for the new user. The JWPTE produces bit estimates for all but the new user (which are discarded) and a phase estimate for the new user. The T_r phase estimates calculated during the refinement stage are averaged to create

$$\hat{\phi}_r = \sum_{i=T_s}^{T_s+T_r-1} \hat{\phi}_k[i]/T_r$$

which is passed to a combiner. The combiner calculates an appropriately weighted average of $\hat{\phi}_s$ and $\hat{\phi}_r$ to produce the final phase estimate, $\hat{\phi}_k$, for the new user.[4] The final phase estimate is passed to the coherent TJD and the detector switches back to the standard mode at $i = T_s + T_r$.

The complexity of this two-part phase estimation procedure per bit is approximately twice the complexity of the JWPTE, hence, it is also twice the complexity of the simple two-part average procedure described in the previous section.

The number of symbols, T_s, over which the new user must send a training weight and over which the detector will remain in the seed mode can be set using a lookup table indexed by E_b/N_o with entries pre-calculated from simulations and experiments prior to system deployment.

The two-part average was tested via simulation using the same scenario as used to test the simple average procedure. A two-part procedure with only 6 training symbols and approximately 45 more symbols needed for refinement in which the new user may send data symbols was capable of achieving a standard deviation of 3.9° for the new user's phase estimate. Another assignment of T_s and T_r were also simulated. For $T_s = 10$ and $T_r = 35$, a phase error standard deviation of 3.9° was also obtained.

The simple averaging procedure described in the previous section can be thought of as the two-part procedure with $T_s = T$ and $T_r = 0$. The simulation results of both the simple averaging procedure and the two part averaging procedure are summarized in Table 3. Notice that benefit to using the two-part averaging procedure in our example (with $T_s = 6, T_r = 45$ or $T_s = 10, T_r = 35$) over the simple averaging procedure (with $T_s = 39, T_r = 0$) is that the new user may begin to send data after only 6 or 10 training symbols as opposed to 39. Both methods, however, allow no new users to enter the system for a comparative number of bit symbols for our example at $E_b/N_o = 6.2$dB. In

---

[3] Recall that the PC-JWPTE allows the uncertain-phase user to send information weights by taking into account the standard deviation in the error between the assumed phase and actual phase for this user.

[4] Note that the phase estimates produced during the refinement stage are dependent upon the phase estimates made during the seed stage, hence the resulting estimate, $\hat{\phi}_k$, is not a weighted sum of i.i.d. random variables.

Table 3: Required duration of the seed and refinement modes for the simple two-part averaging procedure simulated for the acquisition of user 17's phase in the 21 user unit energy minimum distance tree-structured BPSK MA system at $E_b/N_o = 6.2$ dB and a phase mismatch standard deviation $\sigma_m = 4°$ on all other users. Recall that the one-shot phase estimate for this example has a standard deviation of 24°. The seed estimate produced at the end of the seed stage has a standard deviation denoted by σ_s and the final phase estimate standard deviation is denoted by σ_e.

T_s	σ_s	T_r	σ_e
6	10.6°	45	3.9°
10	8.2°	35	3.9°
39	3.9°	0	3.9°

the case of the simple average, users must wait 39 symbols between entering the systems whereas in the case of the two-part procedure user must wait $T_s + T_r = 51$ or 50 symbols. More investigation is necessary to determine if this similarity in waiting periods for the two methods persists for different tree structures and other signal to noise ratios.

During the 6 symbols needed for the seed stage of the acquisition process for our example scenario at $E_B/N_o = 6.2$ dB and a phase mismatch on all users of $\sigma_e = 4°$, the other users experience the BER losses shown by comparing the "Standard mode" and "seed mode" columns of Table 2. For the simple two-part averaging procedure we are also concerned about the BER losses experienced during the 45 symbol refinement stage. For our example, these BERs are shown in the "refinement mode" column of Table 2. Taking into account the error bars on these estimates of 9-13%, the BER in the refinement stage has recovered to that of the coherent BER values for these affected users.

5 Conclusion

The coherent tree-joint detector approach to increasing wireless communication throughput proposed in [3] has been extended for non-coherent operation. This approach relies on having a tree-structure cross correlation matrix corresponding to the set of user signature waveforms. Such a structure, for example, would result from the co-placement of many FDMA users.

Two types of joint demodulator/phase estimation procedures were developed for the case in which users are added to the system one at a time. The first procedure is a one-shot, low complexity, optimal joint detector that demodulates all active users while estimating the phase of the new user to join the system. The complexity for the case of arbitrary interference among users is exponential in the number of users. For the special case of tree structured cross correlations, the complexity of the optimal joint demodulation/phase estimation procedure is reduced to a low order polynomial in most cases.

The second type of joint phase weight estimation procedure increases accuracy of the phase estimate by incorporating many consecutive symbols. Since the optimal multi-shot joint demodulator/phase estimator is of irreducible complexity even for the case of tree structured cross correlations, two sub-optimal, low complexity multi-symbol non-coherent joint demodulation procedures are proposed and analyzed via simulation. The first requires that each new user transmits a full training sequence for phase acquisition, while the second procedure requires only a fraction of a training sequence. Both are capable of achieving near-optimal bit error rates along with high quality phase estimates with short to average training lengths. For example, simulations done with typical SNRs resulted in standard deviations on phase estimates of 4° with training sequences ranging from 5 to 40 bits. No error correction coding was employed.

References

[1] J. G. Proakis. *Digital Communications.* McGraw-Hill, 1989.

[2] S. Verdú. "Computational complexity of optimum multiuser detection". In *Algorithmica.* Springer-Verlag., 1989.

[3] R. E. Learned, A. S. Willsky, and D. M. Boroson. "Low Complexity Optimal Joint Detection for Over-saturated Multiple Access Communications". *IEEE Trans. Signal Processing*, 45(1), Jan. 1997.

[4] A. Duel-Hallen. "Decision-feedback multiuser detector for synchronous code-division multiple access channel". *IEEE Trans. on Comm.*, 41, Feb. 1993.

[5] M. Varanasi and B. Aazhang. "Multistage detection in asynchronous code-division multiple-access communications". *IEEE Trans. on Comm.*, 38, Apr. 1990.

[6] C. Sankaran and A. Ephremides. Optimum multiuser detection with polynomial complexity. In *International Symposium on Information Theory*, Cambridge, MA, Aug. 1998.

[7] S. Verdú. *Multiuser detection.* Cambridge University Press, UK, 1998.

[8] R. E. Learned, S. Mallat, B. Claus, and A. S. Willsky. "Geometric Interpretation of Multiaccess Joint Detection and the Alternating Projection Algorithm". In *Proc. IEEE Conf. Acoust., Speech, Signal Processing*, May 1995.

[9] J. A. F. Ross and D. P. Taylor. "Multiuser signaling in the symbol-synchronous AWGN channel". *IEEE Trans. Inform. Theory*, 41, Jul. 1995.

[10] R. E. Learned. *"Low complexity non-coherent near-optimal multiuser detection for over-saturated MA communication"*. PhD thesis, Massachusetts Institute of Technology: LIDS report LIDS-P-2449, Cambridge, MA, April, 1999.

[11] A. J. Viterbi. "Phase-locked loop dynamics in the presence of noise by Fokker-Plank techniques". *Proc. IEEE*, December 1963.

[12] R. E. Learned. *"Low complexity optimal joint detection for over-saturated multiple access communication"*. PhD thesis, Massachusetts Institute of Technology, Cambridge, MA, January, 1997.

Adaptive Multiuser Decision Feedback for Asynchronous Cellular DS-CDMA

Rapeepat Ratasuk ‡ Graeme Woodward † Michael L. Honig ‡
ratasuk@ece.nwu.edu graemew@southern-poro.com mh@ece.nwu.edu

‡ Dept. of ECE
Northwestern University
2145 Sheridan Rd
Evanston, IL, 60208

† Southern Poro Communications
Annandale (Sydney), NSW
Australia

Abstract

We consider adaptive demodulation of multiple users in an asynchronous DS-CDMA system with frequency-selective fading and inter-cell interference. Two types of demodulation techniques are presented (successive and parallel) along with their Least Squares adaptive implementations. For frequency-selective fading channels, we present an adaptive structure based on embedding the received vectors in a higher-dimensional space. In this case, a training sequence is necessary to estimate the coefficients. If the intra-cell users' signatures and channels are known, the filter coefficients may be estimated without a training sequence. Simulation results show that the adaptive DFDs in frequency-selective channels offers significant improvement over the Least-Squares linear receiver with comparable complexity. Convergence of the DFDs is comparable to that of the linear LS receiver.

1 Introduction

We consider adaptive multiuser demodulation of digital signals in a cellular environment. The approach we take is to use multiuser decision-feedback demodulation which incorporates interference cancellation to improve performance. Decision-feedback equalization for multiple-input/multiple output (MIMO) channels was presented in [1], and applied to multiuser detection for DS-CDMA in [2, 3, 4, 5].

The emphasis of our study is on an *adaptive* multiuser decision-feedback demodulator (DFD), which promises superior performance to an adaptive linear Minimum Mean-Square Error (MMSE) receiver, at the cost of a moderate increase in complexity. The feedforward filter of the adaptive DFD contains an adaptive *linear* MMSE filter while the feedback filter is used to cancel interference. Two types of cancellation techniques are considered: successive and parallel. The Successive DFD (S-DFD) cancels users successively in order of decreasing received power. This is to reduce the effect of error propagation so that weak users benefit from reliable cancellation of stronger users. The Parallel DFD (P-DFD) attempts to cancel all interference simultaneously. When the initial bit estimates are reliable, this technique ensures a greater degree of interference cancellation.

For the channel model considered, both multiple-access and intersymbol interference are present. That is, all users are subject to slow frequency-selective fading. In addition, the system is assumed to be asynchronous. As a result, the optimal filters have an infinite impulse response [3]. In this paper, we approximate the filters as finite impulse response filters (see [6, 7]) by embedding the received vectors in a higher-dimensional space. As a result, intersymbol interference can be treated as interference from separate, virtual users. Once the solution to this higher dimensional minimization problem is found, we can easily extract the solution to the original (lower-dimensional) problem.

In the next section, we present the communication system model under consideration. In Section 3, we derive the MMSE-DFD for asynchronous and frequency-selective fading channels. This is followed, in Section 4, by a presentation of the adaptive algorithms for estimating the filter coefficients. In Section 5, we present the adaptive DFDs with inter-cell interference followed by a discussion on blind estimation in Section 6. Simulation results are then presented in Section 7, and conclusions are presented in Section 8.

2 System Model

The communication scenario considered is asynchronous DS-CDMA with frequency-selective fading channels. The transmitted symbols are offset by multiples of a chip limited to $\pm\frac{T}{2}$. Thus, during a particular observation window, we observe contributions from the current as well as previous and subsequent symbols. The received vector at time i can be written as

$$\mathbf{r}(i) = \begin{bmatrix}\mathbf{S}_{-1} & \mathbf{S}_0 & \mathbf{S}_{+1}\end{bmatrix} \begin{bmatrix}\mathbf{b}(i-1)\\ \mathbf{b}(i)\\ \mathbf{b}(i+1)\end{bmatrix} + \mathbf{n}(i) \tag{1}$$

where $\begin{bmatrix}\mathbf{S}_{-1} & \mathbf{S}_0 & \mathbf{S}_{+1}\end{bmatrix}$ comprises the shifted spreading codes corresponding to the previous, current, and next transmitted symbols. The noise $\mathbf{n}(i)$ is complex-value with covariance $\sigma_n^2\mathbf{I}$. When multipath is also considered, the received vector can be expressed as

$$\mathbf{r}(i) = \mathbf{H}\begin{bmatrix}\mathcal{S}_1 & \mathcal{S}_2 & \ldots & \mathcal{S}_{L+1} & \mathcal{S}_{L+2}\end{bmatrix} \begin{bmatrix}\mathbf{b}(i-L)\\ \vdots\\ \mathbf{b}(i)\\ \mathbf{b}(i+1)\end{bmatrix} + \mathbf{n}(i) \tag{2}$$

where $\mathbf{H}$ is the fading channel matrix,

$$\mathcal{S}_l = \begin{cases}\mathbf{S}_{-1}, & l=1\\ \mathbf{S}_{-1}+\mathbf{S}_0, & l=2\\ \mathbf{S}_{-1}+\mathbf{S}_0+\mathbf{S}_{+1}, & 3<l<L\\ \mathbf{S}_0+\mathbf{S}_{+1}, & l=L+1\\ \mathbf{S}_{+1}, & l=L+2\end{cases} \tag{3}$$

and L is the number of paths. For the numerical results that follow, all channels are assumed to be constant throughout each packet, but are statistically independent from packet to packet.

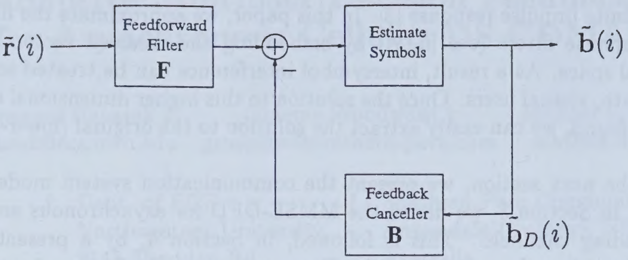


Figure 1: Decision-Feedback Demodulator Block Diagram

In order to capture the energy of the transmitted symbols, it is necessary to expand the size of our observation window by stacking the received vectors as

$$\tilde{\mathbf{r}}(i) = \tilde{\mathbf{P}}\tilde{\mathbf{b}}(i) + \tilde{\mathbf{n}}(i) \tag{4}$$

where

$$\begin{aligned} \tilde{\mathbf{r}}(i) &= \left[\mathbf{r}(i-1) \quad \ldots \quad \mathbf{r}(i+L)\right]^T, \quad \tilde{\mathbf{b}}(i) = \left[\mathbf{b}(i-L-1) \quad \ldots \quad \mathbf{r}(i+L+1)\right]^T, \\ \tilde{\mathbf{n}}(i) &= \left[\mathbf{n}(i-1) \quad \ldots \quad \mathbf{n}(i+L)\right]^T \end{aligned} \tag{5}$$

and $\tilde{\mathbf{P}}$ is the overall channel response. In order to simplify the notation, we will drop the tilde from the variables in the remainder of this paper. Any variable referred to in the remainder of this paper is in the stacked form as shown in (5).

3 MMSE-DFD

Given the structure of the Decision Feedback Demodulator in Figure 1, the goal is to derive the feedforward ($\mathbf{F}_k$) and feedback ($\mathbf{B}_k$) filters to minimize the minimum mean squared error (MSE). That is, we wish to derive $\mathbf{F}_k$ and $\mathbf{B}_k$ to

$$\min_{\mathbf{F}_k,\mathbf{B}_k} \mathrm{MSE}_k = \min_{\mathbf{F}_k,\mathbf{B}_k} E\left[\left|\mathbf{b}_k(i) - \left(\mathbf{F}_k\mathbf{r}(i) - \mathbf{B}_k\mathbf{b}_D(i)\right)\right|^2\right] \tag{6}$$

where $\mathbf{b}_D(i)$ comprises a subset of decisions to be fed back.

Following the discussion in [8], we may minimize (6) by expressing the transmitted bits as

$$\mathbf{r}(i) = \mathbf{P}_D\mathbf{b}_D(i) + \mathbf{P}_U\mathbf{b}_U(i) + \mathbf{n}(i) \tag{7}$$

where $\mathbf{P}_D$ and $\mathbf{P}_U$ are the signature matrices corresponding to the detected and undetected users, respectively. The mean squared error is then given by

$$\mathrm{MSE}_k = 1 + \left(\mathbf{B}_k - \mathbf{F}_k\mathbf{P}_D\right)\left(\mathbf{B}_k - \mathbf{F}_k\mathbf{P}_D\right)^\dagger + \mathbf{F}_k\mathbf{R}_U\mathbf{F}_k^\dagger - \left(\mathbf{F}_k\mathbf{p}_k + \mathbf{p}_k^\dagger\mathbf{F}_k^\dagger\right) \tag{8}$$

where $\mathbf{p}_k$ is the signature vector, including the channel, associated with user k and

$$\mathbf{R}_U = \mathbf{P}_U\mathbf{P}_U^\dagger + \sigma_n^2\mathbf{I} \tag{9}$$

is the correlation matrix of the undetected users plus noise. From (8), it is straightforward to show that the value of $\mathbf{B}_k$ which minimizes the MSE is

$$\mathbf{B}_k = \mathbf{F}_k \mathbf{P}_D \tag{10}$$

The optimal feedforward filter is then easily found to be

$$\mathbf{F}_k = \mathbf{p}_k^{\dagger} \mathbf{R}_U^{-1} \tag{11}$$

In the S-DFD, users are cancelled successively. That is, only symbols from users previously detected may be used in the cancellation process. Hence, for user k,

$$\mathbf{P}_D = \begin{bmatrix} \mathbf{p}_1 & \mathbf{p}_2 & \cdots & \mathbf{p}_{k-1} \end{bmatrix}, \quad \mathbf{P}_U = \begin{bmatrix} \mathbf{p}_{k+1} & \mathbf{p}_{k+2} & \cdots & \mathbf{p}_K \end{bmatrix} \tag{12}$$

While this approach limits the deleterious effect of error propagation, performance of the first few users may be poor due to lack of cancellation. The P-DFD attempts to cancel interference from all other users simultaneously. For user k,

$$\mathbf{P}_D = \begin{bmatrix} \mathbf{p}_1 & \cdots & \mathbf{p}_{k-1} & \mathbf{p}_{k+1} & \cdots & \mathbf{p}_K \end{bmatrix}, \quad \mathbf{P}_U = \mathbf{p}_k \tag{13}$$

This scheme offers a greater degree of interference cancellation, but also suffers more from error propagation.

4 Adaptive DFD

When channel parameters are unknown, a training sequence may be used to estimate the filter coefficients. These estimates can be obtained by minimizing the least squares (LS) cost function

$$\min_{\hat{\mathbf{F}},\hat{\mathbf{B}}} \mathcal{E}_{ls} = \min_{\hat{\mathbf{F}},\hat{\mathbf{B}}} \frac{1}{B} \sum_{i=1}^{B} \beta^{B-i} \left| \tilde{\mathbf{b}}(i) - (\hat{\mathbf{F}}\tilde{\mathbf{r}}(i) - \hat{\mathbf{B}}\hat{\tilde{\mathbf{b}}}(i)) \right|^2 \tag{14}$$

where B is the number of bits in a packet and β is the discount factor used in non-stationary environments. Note that the $M \times 1$ (where $M = K(L+2)$) stacked vector $\tilde{\mathbf{b}}(i)$ represents the system embedded in a higher-dimensional space. The intersymbol interference is then treated as interference from separate, virtual users.

It is straightforward to show that the solution to the lower dimensional problem can be easily extracted from (14). Specifically, the desired $\mathbf{b}(i)$ sub-block of $\tilde{\mathbf{b}}(i)$ is contained in the higher-dimensional minimization problem as

$$\min_{\hat{\mathbf{F}},\hat{\mathbf{B}}} \frac{1}{B} \sum_{i=1}^{B} \beta^{B-i} \left| \mathbf{b}(i) - \hat{\mathbf{F}}_{\{(L-1)K+1):(LK)\}} \tilde{\mathbf{r}}(i) + \hat{\mathbf{B}}_{\{(L-1)K+1):(LK)\}} \tilde{\mathbf{b}}(i) \right|^2 \tag{15}$$

where $\hat{\mathbf{F}}_{\{(L-1)K+1):(LK)\}}$ and $\hat{\mathbf{B}}_{\{(L-1)K+1):(LK)\}}$ denote the $((L-1)K+1)^{th}$ to $(LK)^{th}$ rows of $\hat{\mathbf{F}}$ and $\hat{\mathbf{B}}$, respectively. Thus, our desired solution is embedded in the solution to the higher-dimensional problem and can be extracted as

$$\hat{\mathbf{F}}_{desired} = \hat{\mathbf{F}}_{\{(L-1)K+1):(LK)\}} \tag{16}$$

$$\hat{\mathbf{B}}_{desired} = \hat{\mathbf{B}}_{\{(L-1)K+1):(LK)\}} \tag{17}$$

In the absence of feedback error, $\hat{\mathbf{b}}(i) = \mathbf{b}(i)$, (14) evaluates to

$$\min_{\hat{\mathbf{F}},\hat{\mathbf{B}}} \mathcal{E}_{ls} = \min_{\hat{\mathbf{F}},\hat{\mathbf{B}}} \frac{1}{B} \sum_{i=1}^{B} \beta^{B-i} \Big[(\mathbf{I}+\hat{\mathbf{B}})\mathbf{b}(i)\mathbf{b}^{\dagger}(i)(\mathbf{I}+\hat{\mathbf{B}})^{\dagger} + \hat{\mathbf{F}}\mathbf{r}(i)\mathbf{r}^{\dagger}(i)\hat{\mathbf{F}} \\ - (\mathbf{I}+\hat{\mathbf{B}})\mathbf{b}(i)\mathbf{r}^{\dagger}(i)\hat{\mathbf{F}} - \hat{\mathbf{F}}\mathbf{r}(i)\mathbf{b}^{\dagger}(i)(\mathbf{I}+\hat{\mathbf{B}})^{\dagger} \Big] \tag{18}$$

The above minimization may be solved via the orthogonality principle to yield

$$\hat{\mathbf{F}} = (\mathbf{I}+\hat{\mathbf{B}})\hat{\mathbf{P}}^{\dagger}\hat{\mathbf{R}}^{-1}, \tag{19}$$

where

$$\hat{\mathbf{R}} = \frac{1}{B}\sum_{i=1}^{B} \mathbf{r}(i)\mathbf{r}^{\dagger}(i), \qquad \hat{\mathbf{P}} = \frac{1}{B}\sum_{i=1}^{B} \mathbf{r}(i)\mathbf{b}^{\dagger}(i) \tag{20}$$

are the time-averaged correlation matrix and the time-averaged cross-correlation matrix, respectively.

The feedforward filter $\hat{\mathbf{F}}$ can be separated into a linear LS filter $\hat{\mathbf{P}}^{\dagger}\hat{\mathbf{R}}^{-1}$, followed by the filter $(\mathbf{I}+\hat{\mathbf{B}})$. Since the front-end filter is based on the linear LS receiver, it is robust with respect to uncancelled interference (including other-cell users). In addition, when error propagation is severe, the feedback filter may be disabled and decisions can be taken from the linear LS receiver. Note that the derivation is independent of the cancellation structure. Thus, (19) is the same for both P-DFD and S-DFD.

Substituting (19) into (14) gives the LS cost function

$$\mathcal{E}_{ls} = (\mathbf{I}+\hat{\mathbf{B}})\hat{\mathbf{Q}}(\mathbf{I}+\hat{\mathbf{B}})^{\dagger} \tag{21}$$

where $\hat{\mathbf{Q}}$ is the error covariance of the linear LS receiver, given by

$$\hat{\mathbf{Q}} = \frac{1}{B}\sum_{i=1}^{B} \left(\mathbf{b}(i) - \hat{\mathbf{P}}^{\dagger}\hat{\mathbf{R}}^{-1}\mathbf{r}(i)\right)\left(\mathbf{b}(i) - \hat{\mathbf{P}}^{\dagger}\hat{\mathbf{R}}^{-1}\mathbf{r}(i)\right)^{\dagger} \tag{22}$$

In addition, restrictions may be placed on the structure of $\hat{\mathbf{B}}$, corresponding to the cancellation technique. The solution to this minimization problem then varies according to the cancellation method.

4.1 Successive DFD

Since the interference is to be cancelled successively, the feedback matrix $\hat{\mathbf{B}}$ is lower diagonal with ones on the diagonal. The kth row is

$$\hat{\mathbf{B}}_k = \begin{bmatrix} \hat{\mathbf{B}}_{k,1:k-1} & 1 & 0 & \ldots & 0 \end{bmatrix} \tag{23}$$

where $\hat{\mathbf{B}}_{k,1:k-1}$ is a vector containing elements $1, \ldots, k-1$ of the kth row of $\hat{\mathbf{B}}$. The minimization problem of (21) then reduces to

$$\mathcal{E}_{ls} = \hat{\mathbf{B}}_{k,1:k-1}\hat{\mathbf{Q}}_{1:k-1,1:k-1}\hat{\mathbf{B}}^{\dagger}_{k,1:k-1} + \hat{\mathbf{q}}_{k,1:k-1}\hat{\mathbf{B}}^{\dagger}_{k,1:k-1} \\ + \hat{\mathbf{B}}_{k,1:k-1}\hat{\mathbf{q}}^{\dagger}_{k,1:k-1} + \hat{\mathbf{Q}}_{k,k} \tag{24}$$

where $\hat{\mathbf{q}}_k$ is the entire kth row of $\hat{\mathbf{Q}}$ with the kth element removed. The solution is then given by

$$\hat{\mathbf{B}}_{k,1:k-1} = \hat{\mathbf{q}}^{\dagger}_{k,1:k-1}\hat{\mathbf{Q}}^{-1}_{1:k-1,1:k-1} \tag{25}$$

Note that the coefficients of the feedback filter are calculated successively from user 1 to k. For user k, $k-1$ coefficients are to be estimated. Hence, convergence is non-uniform with faster convergence for users that are early in the detection order. This non-uniformity, however, is beneficial since it ensures reliable symbol estimates will be available for cancellation by undetected users.

The backward coefficients derived in (25) are identical to those obtained by whitening the matrix error covariance $\hat{\mathbf{Q}}$ [3]. That is,

$$\hat{\mathbf{B}} = \hat{\mathbf{L}}^{-1} - \mathbf{I} \tag{26}$$

where $\hat{\mathbf{L}}$ is defined by the Cholesky factorization

$$\hat{\mathbf{Q}} = \hat{\mathbf{L}}\hat{\mathbf{D}}\hat{\mathbf{L}}^{\dagger} \tag{27}$$

Notice that this is the same as for synchronous CDMA [9] where the matrices $\mathbf{R}$ and $\mathbf{P}$ are defined in the analogous way.

4.2 Parallel DFD

For the parallel DFD, the lower-diagonal constraint on the feedback filters is removed, hence tentative symbols from all other users are used simultaneously in the cancellation process. Thus, we may write the kth row of $\hat{\mathbf{B}}$ as

$$\hat{\mathbf{B}}_k = \left[\hat{\mathbf{B}}_{k,1:k-1} \quad 1 \quad \hat{\mathbf{B}}_{k,k+1:K}\right] \tag{28}$$

By defining

$$\hat{\mathcal{B}}_k = \left[\hat{\mathbf{B}}_{k,1:k-1} \quad \hat{\mathbf{B}}_{k,k+1:K}\right] \tag{29}$$

we may express (21) as

$$\mathcal{E}_{ls} = \hat{\mathcal{B}}_k\hat{\mathcal{Q}}_k\hat{\mathcal{B}}_k^{\dagger} + \hat{\mathbf{q}}_k\hat{\mathcal{B}}_k^{\dagger} + \hat{\mathcal{B}}_k^{\dagger}\hat{\mathbf{q}}_k^{\dagger} + \hat{\mathbf{Q}}_{k,k} \tag{30}$$

where

$$\hat{\mathcal{Q}}_k = \begin{bmatrix} \hat{\mathbf{Q}}_{1:k-1,1:k-1} & \hat{\mathbf{Q}}_{1:k-1,k+1:K} \\ \hat{\mathbf{Q}}_{k+1:K,1:k-1} & \hat{\mathbf{Q}}_{k+1:K,k+1:K} \end{bmatrix} \tag{31}$$

and $\hat{\mathbf{q}}_k$ is given in the previous section. The minimizing filter coefficients are given by

$$\left[\hat{\mathbf{B}}_{k,1:k-1} \quad \hat{\mathbf{B}}_{k,k+1:K}\right] = \hat{\mathbf{q}}_k^{\dagger}\hat{\mathcal{Q}}_k^{-1} \tag{32}$$

There are $K-1$ coefficients associated with each user. Convergence is therefore uniform for all users and is equivalent to that of the final user for the S-DFD.

For the P-DFD, the backward filter coefficients can be interpret as an *error interpolation* filter. Specifically, the kth row of $\mathbf{I} + \hat{\mathbf{B}}$ is given by the vector $\hat{\mathcal{B}}_k$ which minimizes the least-squares interpolation error

$$\xi_k = \frac{1}{B}\sum_{i=1}^{B}\left|\epsilon_k - \hat{\mathcal{B}}_k\tilde{\epsilon}_k\right|^2 \tag{33}$$

where

$$\epsilon_k = \frac{1}{B}\sum_{i=1}^{B}\left[\mathbf{b}_k(i) - \hat{\mathbf{p}}_k^{\dagger}\hat{\mathbf{R}}\mathbf{r}(i)\right] \tag{34}$$

and $\tilde{\epsilon}_k$ is the vector ϵ_k with the kth component removed.

5 Inter-cell Interference

In a cellular system, both intra-cell and inter-cell users are present. While intra-cell users can be cancelled, the symbols of the inter-cell users are usually not estimated and, hence, the associated interference cannot be cancelled. The received vector is written as

$$\mathbf{r}(i) = \mathbf{P}\mathbf{b}(i) + \mathbf{P}_I\mathbf{b}_I(i) + \mathbf{n}(i) \tag{35}$$

where $\mathbf{P}_I$ and $\mathbf{b}_I(i)$ are the overall channel matrix and transmitted bits of the inter-cell interferers, respectively. It is straightforward to show that the derivation of the LS DFDs in the presence of inter-cell interference is the same as before.

6 Blind Algorithm

In the absence of multipath, if the signatures of the intra-cell users are known, then we can use this knowledge to estimate the filter coefficients without a training sequence. This reduces the overhead cost of transmitting a training sequence. Specifically, the front-end linear filter is given by

$$\hat{\mathbf{C}} = \hat{\mathbf{R}}^{-1}\mathbf{P} \tag{36}$$

where $\mathbf{P}$ contains the *known* signatures. The feedback filter depends only on the error covariance of the linear LS receiver. Hence, our blind approach is based on estimating this covariance matrix. If the transmitted bits are independently and identically distributed then the error covariance can be approximated as

$$\hat{\mathbf{Q}} = \mathbf{I} - \mathbf{P}^{\dagger}\hat{\mathbf{R}}^{-1}\mathbf{P} \tag{37}$$

and the feedback filter can be estimated via either (25) or (32).

While this unsupervised adaptation is appealing, it converges slowly. This renders the method inadequate in rapidly changing mobile environments. In addition, when the number of received vectors is small, the matrix $\hat{\mathbf{R}}$ is ill-conditioned, which introduces numerical problems. This problem may be rectified by adding a small constant to the diagonal terms of the covariance matrix. Alternately, the feedback matrix may be disabled, and estimates taken directly from the linear front-end, until a sufficient number of samples is obtained.

7 Numerical Results

Our numerical results illustrate the performance of the adaptive DFDs with 4 users and a spreading gain of 8 under frequency selective channels and AWGN. The channel is assumed to consist of three independent paths with power profile [0dB -2dB -4dB] and a separation of one chip interval per path. For the results, the filters are first trained with the specified number of training bits. These resulting filter coefficients are then used to compute the error rate for a block of 150 additional symbols and averaged over sufficient number of independent runs (typically over 10,000). The initial bit estimates used for cancellation are taken from the linear LS receiver. To alleviate the effect of error propagation, we introduced erasure into our initial bit estimates. In this case, if the magnitude of the real part of the decision variable is less than a prescribed value α, a zero is inserted instead. Both intra-cell and inter-cell (4 surrounding cells) interference are considered. Adjacent cell users are power-controlled by adjacent cell base stations (see [10]).

Figure 2 illustrates error rate as a function of training data for the adaptive receivers. Due to the large number of filter coefficients, the decision-directed DFDs converge slowly, requiring approximately 500 training bits. For the linear LS and adaptive P-DFD receivers, the error rate is averaged over all users. For the adaptive S-DFD, only the performance of the weakest and strongest users are shown since the users are demodulated according to received power. It is apparent that a substantial performance improvement in BER relative to the linear LS receiver can be obtained even at partial convergence. In addition, performance degrades gracefully in the presence of inter-cell interference.

Figure 3 shows the performance of the adaptive DFDs as a function of $\frac{E_b}{N_0}$. In this case, a training sequence of 1,000 symbols is used to estimate the filter coefficients. Again, the DFDs outperforms the linear receiver consistently. The P-DFD usually provides the best performance over all receivers. This is especially true at high $\frac{E_b}{N_0}$'s where initial bit estimates are highly reliable.

Finally, Figure 4 shows the convergence of the blind algorithm where the users' signatures and channels are known. Note that the plot shows performance for 100-5,000 training bits. The blind algorithm converges rather slowly, requiring approximately 3,500 training symbols. Due to its slow convergence, the blind algorithm is more suitable for stationary or slowly-varying channels.

8 Conclusions

We have studied adaptive DFD receivers for an asynchronous DS-CDMA system with frequency-selective channels. It has been shown that for the cases considered, the DFDs offer a significant performance improvement over the adaptive linear receiver with comparable complexity. As the number of DFD coefficients increases, more training data is needed to obtain an accurate estimate of the MMSE coefficients, which can limit the application of the adaptive DFDs to slowly-varying channels.

References

[1] A. Duel-Hallen, "Equalizers for Multiple Input/Multiple Output Channels and PAM Systems with Cyclostationary Input Sequences," *IEEE Journal on Selected Areas in Communications*,

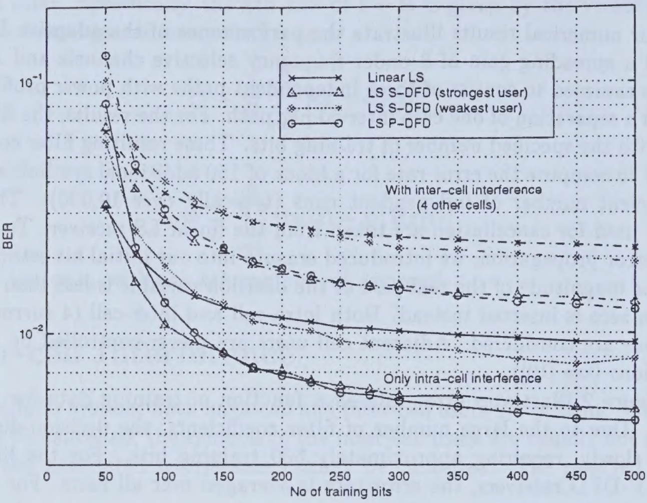


Figure 2: Convergence of adaptive DFDs for asynchronous DS-CDMA with Rayleigh fading channel (3 paths): 4 users, N=8, erasure with $\alpha = 0.17$.

vol. 10, pp. 630–639, Apr. 1992.

[2] A. Duel-Hallen, "Decorrelating Decision-Feedback Multiuser Detector for Synchronous Code-Division Multiple Access Channel," *IEEE Transactions on Communications*, vol. 41, pp. 285–290, Feb. 1993.

[3] A. Duel-Hallen, "A Family of Multiuser Decision-Feedback Detectors for Asynchronous Code-Division Multiple-Access Channels," *IEEE Transactions on Communications*, vol. 43, pp. 421–434, feb/mar/apr 1995.

[4] A. Klein, G. K. Kaleh, and P. W. Baier, "Zero Forcing and Minimum Mean-Square-Error Equalization for Multiuser Detection in Code-Division Multiple-Access," *IEEE Transactions on Vehicular Technology*, vol. 45, pp. 276–287, May 1996.

[5] A. Host-Madsen and K. S. Cho, "MMSE/PIC Multiuser Detection for DS/CDMA Systems with Inter- and Intra-cell Interference," *IEEE Transactions on Communications*, vol. 47, pp. 291–299, Feb. 1999.

[6] C. Belfoire and J. Park, Jr., "Decision Feedback Equlization," *Proceedings of the IEEE*, vol. 67, pp. 1143–1156, Aug. 1979.

[7] N. Al-Dhahir and J. M. Cioffi, "MMSE Decision-Feedback Equalizers: Finite-Length Results," *IEEE Transactions on Information Theory*, vol. 41, pp. 961–975, July 1995.

[8] P. B. Rapajic, M. L. Honig, and G. Woodward, "Multiuser Decision-Feedback Detection: Performance Bounds and Adaptive Algorithms," *IEEE International Symposium on Information Theory*, Aug. 1998.

[9] G. Woodward, R. Ratasuk, and M. Honig, "Multistage Multiuser Decision Feedback Detection for DS-CDMA ," *International Conference on Communications*, June 1999.

[10] M. L. Honig, "Adaptive Linear Interference Suppression for Packet DS-CDMA," *European Transaction on Telecommunications*, vol. 9, pp. 1–9, March-April 1998.

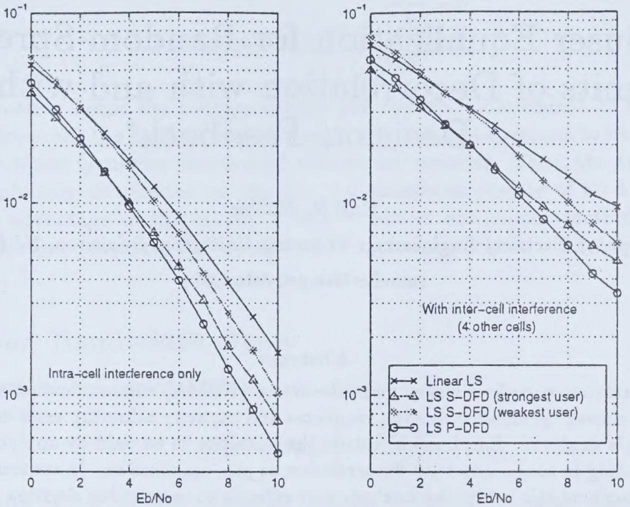


Figure 3: Performance of the DFD receivers for asynchronous DS-CDMA with Rayleigh fading channel (3 paths): 4 users, N=8, erasure with $\alpha = 0.17$.

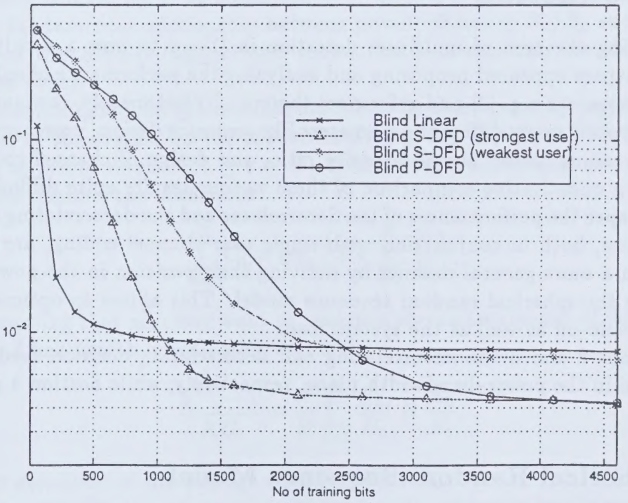


Figure 4: Convergence of blind DFDs for asynchronous DS-CDMA with Rayleigh fading channel (3 paths): 4 users, N=8, erasure with $\alpha = 0.17$.

Multiuser Equalization for Random Spreading: Limits of Decorrelation with and without Decision–Feedback*

Ralf R. Müller
Dept. of Electrical Engineering, Princeton University, Princeton, NJ 08540
rmueller@ee.princeton.edu

Abstract

Synchronous code–division multiple–access (CDMA) communication with randomly chosen spherical spreading sequences and capacity achieving error correction coding is analyzed. Emphasis is put on the penalties to be paid by applying single user coding in conjuction with decorrelation as pre–equalization. In contrast to previous work on this topic, the analysis also extends to re–encoded decision feedback structures where an equal rate and and an equal power case are distinguished. The results are nonasymptotic in the number of users and show that fluctuations of the signal–to–noise ratio due to the random nature of the sequences has no deleterious effect onto capacity for the spherical random sequence model.

1 Introduction

From founding the area of multiuser detection in [1] up to now a nearly uncountable number of papers appeared proposing and analyzing the performance of suboptimal multiuser detectors, see e.g. [2] and references therein. Unfortunately, comparing multiuser detectors involves many different parameters like sequence design, power control, number of users, spreading factor, signal–to–noise ratio, and design of channel coding (if used). This makes a quantitative comparison of these various results quite difficult.

In this paper the performances of the decorrelator and the decorrelating decision–feedback equalizer, both in conjunction with single user channel coding, are analyzed and compared in a more general manner by deriving their position in the power–bandwidth plane under the spherical random sequence model. This allows to optimize the coding spreading trade–off as well as the system load.

In Sections 2 and 3 the decorrelating and decorrelating decision–feedback detector are analyzed in the power–bandwidth plane, respectively, while Section 4 points out the conclusions.

1.1 Spherical Random Sequence Model

In this work each overall spreading sequence $\boldsymbol{b}$ is modelled by an L dimensional complex Gaussian random vector $[g_1, g_2, \ldots, g_L]^{\mathrm{T}}$ which is normalized to unit Euclidian norm with

---

[*] This work was performed when the author was with Telecommunications Institute II, University of Erlangen–Nuremberg, Cauerstr. 7/NT, 91058 Erlangen, Germany.

L denoting the spreading factor

$$b = [g_1, g_2, \dots, g_L]^{\mathrm{T}} \Big/ \sqrt{\sum_{i=1}^{L} |g_i|^2}. \tag{1}$$

This normalization keeps the average energy per symbol independent of the signature sequences. Moreover, the random variables $g_i, 1 \le i \le L$, are assumed to be independent zero–mean complex Gaussian distributed with equal variance. Thus, the spreading vectors b are uniformly distributed on the $L-1$ dimensional surface of an L dimensional sphere. The advantage of this model is that it allows analytical tractability of many multiuser detection schemes. For details on this random sequence model the reader is referred to [2, 3].

1.2 Power–Bandwidth Plane

From Shannon's [4] celebrated formula

$$C_{\mathrm{T}} = B \log_2 \left(1 + \frac{P}{\sigma^2}\right) \tag{2}$$

for the capacity of the channel corrupted by additive white Gaussian noise (AWGN) of power σ^2, limited to bandwidth B, and constrained to signal power P it is staightforward to obtain

$$\frac{E_{\mathrm{b}}}{N_0} = \frac{2^C - 1}{C} \tag{3}$$

with the capacity normalized to bandwidth, the energy per bit, and the noise power spectral density defined by $C \triangleq C_{\mathrm{T}}/B, E_{\mathrm{b}} \triangleq P/C_{\mathrm{T}}, N_0 \triangleq \sigma^2/B$, respectively. Eq. (3) establishes a fundamental relationship between power efficiency N_0/E_{b} and bandwidth–efficiency C of the AWGN channel and shows that one can only be obtained at the expense of the other.

Multiuser communication systems are described by their capacity region [5] which is given by a set of constraints on the capacities $C_i, 1 \le i \le K$, of the K individual users transmitting at power levels $P_i, 1 \le i \le K$. If the total transmission rate is maximum, for the AWGN channel the constraint

$$\sum_{i=1}^{K} C_i = \log_2 \left(1 + \sum_{i=1}^{K} \frac{P_i}{N}\right) \tag{4}$$

bites [5]. Assuming that all users want to transmit the same amount of data within the same time and bandwidth, i.e. $C_i = C, \forall i$, yields

$$\frac{2^{KC} - 1}{KC} = \frac{1}{K} \sum_{i=1}^{K} \frac{E_{\mathrm{b}i}}{N_0} \triangleq \frac{\overline{E_{\mathrm{b}}}}{N_0}, \tag{5}$$

with implicit definition of the average energy per bit $\overline{E_{\mathrm{b}}}$.

Considering CDMA, there is an inherent bandwidth extension by the spreading factor L. Thus, spectral efficiency (total capacity per chip) relates to single–user capacity C (capacity per user and symbol) like

$$\Gamma = \frac{K}{L} C \triangleq \zeta C, \tag{6}$$

cf. [6], with ζ denoting the *load* of the system. In order to illustrate the performance of CDMA in the power–bandwidth plane we need to find a relationship between spectral efficiency Γ and power efficiency $N_0/\overline{E_b}$. Spectral efficiency of the optimum multiuser decoder with optimum sequence design is given by Eq. (5). It serves as an upper bound on performance of the suboptimum detection schemes analyzed in the following.

2 Analysis of Decorrelation

Spectral efficiency of zero–forcing[1] multiuser equalization with random speading sequences has been addressed in [7] and [8] by using bounds on the average signal–to–noise ratio and Monte–Carlo methods. Asymptotic results, i.e. number of users K and spreading factor L grow over all bounds, have recently been given in [3]. For related work on performance of linear multiuser receivers in random environments, see also [9, 10]. In this paper, we also examine the case of finite K and L.

Using random signature sequences the signal–to–noise ratio at decorrelator output is a random variable. Setting $P_i, \forall i$, without loss of generality and for ease of notation, its probability density function was derived for spherical random sequences as [11, 12, 13]

$$\mathrm{p}_{\gamma_{\mathrm{ZF}}}(y) = \begin{cases} \sigma^2 \dfrac{(\sigma^2 y)^{L-K}(1-\sigma^2 y)^{K-2}}{\mathrm{B}(L-K+1, K-1)} & 0 \le y \le \sigma_n^{-2} \\ 0 & \text{otherwise} \end{cases} \tag{7}$$

with $\mathrm{B}(a,b) \triangleq \int_0^1 t^{a-1}(1-t)^{b-1}dt$ denoting the beta function.

Asymptotically, i.e. for $K = \zeta L \to \infty, \zeta \in (0;1]$, the variance of the signal–to–noise ratio vanishes

$$\lim_{K=\zeta L\to\infty} \mathrm{Var}\{\gamma_{\mathrm{ZF}}\} = 0. \tag{8}$$

yielding that the signal–to–noise ratio becomes a deterministic value.

These properties of the signal–to–noise ratio of the decorrelator's output signal are used in the following to give insight into the trade–off between power and bandwidth efficiency of zero–forcing multiuser equalization.

Proposition 1 *The average channel capacity of a CDMA–channel with K users with unit powers which is disturbed by complex AWGN of variance σ^2 and equalized according to the zero–forcing criterion is given by*

$$\overline{C}_{\mathrm{ZF}} = \log_2\left(1+\frac{1}{\sigma^2}\right) - \frac{1}{\ln(2)}\sum_{n=1}^{\infty}\frac{(K-1)_n}{n(L)_n}\left(\frac{1}{1+\sigma^2}\right)^n \tag{9}$$

with $(x)_n \triangleq \prod_{i=0}^{n-1}(x+i)$ denoting the Pochhammer polynomials, and bounded by

$$\log_2\left(1+\frac{L-K}{\sigma^2 L}\right) < \overline{C}_{\mathrm{ZF}} < \log_2\left(1+\frac{L-K+1}{\sigma^2 L}\right) \tag{10}$$

if the spreading sequences are uniformly distributed on the $(L-1)$–dimensional surface of an L–sphere.

---

[1] The terms *decorrelation* and *zero–forcing* are used synonymously throughout this paper.

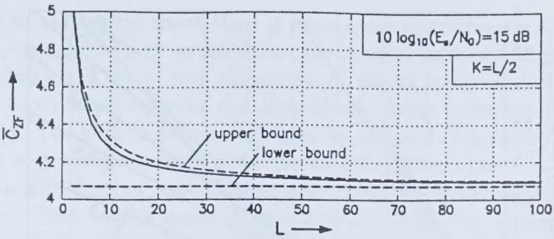


Figure 1: Channel capacity $\overline{C}_{\mathrm{ZF}}$ with zero–forcing equalization versus spreading factor L for $K = L/2$ and $10\log_{10}(E_s/N_0) = 15\,\mathrm{dB}$. The dashed lines refer to the upper and lower bound, respectively.

The proof is omitted in this conference version of the paper. It can be found in [12, 13]. Proposition 1 implies some surprising consequences. Channel capacity is a concave function of the random variable signal–to–noise ratio. Thus, the average capacity is lower than capacity of the average signal–to–noise ratio. Eq. (8) tells that the variance of the signal–to–noise ratio vanishes for infinite spreading factor. This should imply that for infinite L capacity reaches its maximum if K/L is fixed, but surprisingly this statement is disproved by the left hand side of (10). In fact, the average signal–to–noise ratio depends on the $(K-1)/L$ ratio and for fixed $(K-1)/L$ ratio the above statement actually holds, but the small impact by the fact that users do not interfere with themselves overrules the impact of decreasing variance of signal–to–noise ratio. As depicted in Fig. 1 the upper bound is even much closer than the lower bound.

The asymptotic relationship between power and bandwidth efficiency results with (3) and (6)

$$\lim_{K=\zeta L\to\infty}\left(\frac{E_{\mathrm{b}}}{N_0}\right)_{\mathrm{ZF}} = \frac{2^{\Gamma/\zeta}-1}{\Gamma/\zeta-\Gamma} \tag{11}$$

from Proposition 1 and is illustrated in Fig. 2. Eq. (11) does not only hold for signature sequences which are uniformly distributed on the surface of an L-sphere, but also for independent binary distributed signature sequences with zero mean [3]. Fig. 2 indicates that the load ζ has to be carefully adjusted if high spectral efficiency shall be obtained.

Considering Fig. 2, a large gap between the orthogonal system and one applying CDMA with decorrelation and random sequences occurs. Applying (11) the loss of the decorrelator in comparison to the single–user Shannon bound is given by

$$V_{\mathrm{ZF}}(\Gamma) \triangleq \frac{\min\limits_{\zeta}\left(\frac{E_{\mathrm{b}}}{N_0}\right)_{\mathrm{ZF}}}{\frac{2^{\Gamma}-1}{\Gamma}} = \min_{\zeta}\frac{2^{\Gamma/\zeta}-1}{\left(2^{\Gamma}-1\right)(1/\zeta-1)} = \min_{\zeta}\frac{2^{\Gamma(1/\zeta-1)}}{1/\zeta-1} + \frac{\mathcal{O}\left(2^{\Gamma(1/\zeta-2)}\right)}{1/\zeta-1} \tag{12}$$

This simple optimization problem can be easily solved by first order derivation and gives

$$V_{\mathrm{ZF}}(\Gamma) = \mathrm{e}\,\ln(2)\,\Gamma + \mathcal{O}\left(\Gamma 2^{-\Gamma}\right). \tag{13}$$

Obviously, the asymptotic loss, i.e. $V_{\mathrm{ZF}}(\Gamma), \Gamma \to \infty$, is proportional to spectral efficiency. For $\Gamma > 4$ its difference to the exact loss is quite small. Eq. (13) shows that the loss to the optimum detector grows over all bounds if spectral efficiency does. Nevertheless, this is an important improvement in comparison to the conventional detector whose loss becomes infinite for *finite* spectral efficiency because of interference limitation.

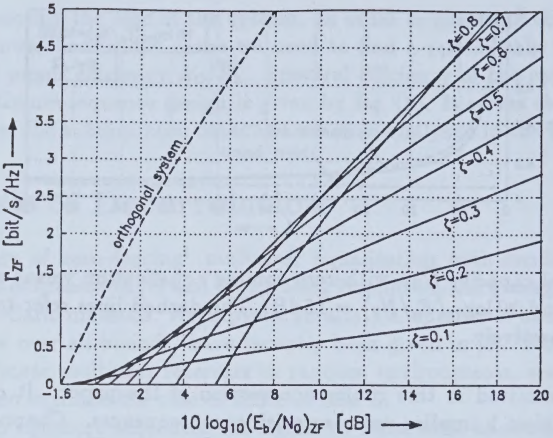


Figure 2: Power–bandwidth–plane for zero–forcing CDMA equalization with the K/L–ratio ζ as parameter. For comparison the dashed line shows an orthogonal system.

3 Decorrelating Decision Feedback Equalization

Combining decorrelating decision–feedback equalizatione with channel coding, there are two main implementations one can think about. The first one, called *direct feedback* in the following, is simply feeding the soft–output signals of the decorrelating decision–feedback detector [2] into the decoders. This method substantially suffers from bit errors. In comparison with uncoded transmission the problem of incorrect decisions is much more severe, as the average energy per symbol is significantly smaller if channel coding is applied. The second one is depicted in Fig. 3 for the two users' case. The cancellation of interfering signals is performed after the decoding process. This provides (almost) error–free feedback.

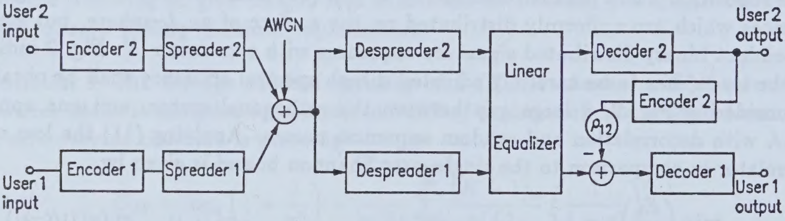

User 2
input
Encoder 2
Spreader 2
AWGN
Despreader 2
Linear
Equalizer
Decoder 2
User 2
output
Encoder 2
ρ_{12}
−
User 1
input
Encoder 1
Spreader 1
Despreader 1
Decoder 1
User 1
output

Figure 3: Structure of decision–feedback decoding for two users.

In this context, that is the evaluation of maximum spectral efficiency, direct feedback is not considered further on, as there are several reasons limiting its sense. The only advantages of direct feedback are its obviously lower complexity, as re–encoding is not required, and the shorter processing delay. In contrast to processing delay, algorithmic delay does not differ in both cases if transmission is at least block–synchronous[2]. This is, as no user can be decoded until the whole codeword has been received. Then, all users can be decoded in arbitrarily short time if processing is arbitrarily fast. As processing may not be fast enough for high speed data transmission and/or a high number of users, the processing delay can be a real advantage of direct feedback. However, for codes operating close to channel capacity the algorithmic delay is expected to dominate the overall data delay. The gain in processing delay by reduced complexity is even negligible, as reencoding is very simple compared to the decoding procedure.

---

[2] That is, the time–shifts between the codewords differ only by few symbol periods.

In contrast to zero–forcing equalization, decorrelating decision–feedback is asymetric in user direction. That is, the users do not operate under the same conditions. Some users are more and some are less disturbed by multiuser interference *in the average.* This fact has severe impact on cellular communications design, see e.g. [13, 14]. It yields that the signals of users transmitting at identical signal–to–noise ratios may be characterized by different signal–to–distortion ratios after multiuser equalization. This implies different average channel capacities among the users. Vice versa, users with equal average channel capacities require different signal–to–noise ratios. Although there are many cases with different joint rate *and* signal–to–noise ratio distributions among the users, we distinguish only the two cases previously mentioned in this paragraph, i.e. the equal rate and the equal power case.

3.1 Equal Powers

In the following we assume that all users are received with identical signal–to–noise ratios, i.e. their ratio of energy per symbol to noise power density is fixed. This enables to obtain the average capacity of decorrelating decision–feedback equalization from averaging the average capacity of the decorrelating detector with k users given by (9), see Proposition 1, over a discrete uniform distribution of users $1 \leq k \leq K$.

Proposition 2 *The average channel capacity of a CDMA–channel with K users of unit power which is disturbed by complex AWGN of variance σ^2 and equalized according to the zero–forcing decision–feedback criterion is given by*

$$\overline{C}_{\mathrm{ZFDF1}} = \log_2\left(1+\frac{1}{\sigma^2}\right) - \frac{K-1}{K\ln(2)}\sum_{n=1}^{\infty}\frac{(K)_n}{n(n+1)(L)_n}\left(\frac{1}{1+\sigma^2}\right)^n \tag{14}$$

and bounded by

$$\frac{K-1}{K}\,\mathrm{f}\left(\frac{K}{L(1+\sigma^2)}\right) > \overline{C}_{\mathrm{ZFDF1}} - \log_2\left(1+\frac{1}{\sigma^2}\right) > \frac{K-1}{K}\,\mathrm{f}\left(\frac{K+1}{L(1+\sigma^2)}\right) \tag{15}$$

with

$$\mathrm{f}(x) \triangleq \left(1-\frac{1}{x}\right)\log_2(1-x) - \frac{1}{\ln(2)} \tag{16}$$

if the spreading sequences are uniformly distributed on the $(L-1)$–dimensional surface of an L–sphere.

The proof is omitted in this conference version of the paper. It can be found in [12, 13].

Unfortunately, (14) does not illustrate the influence of the parameters K, L, σ^2 denoting number of users, spreading factor, and noise variance, respectively, very well. For this purpose, it is more helpful to examine the asymptotic behavior, i.e. the number of users grows over all bounds, but the load $\zeta \triangleq K/L$ remains constant. Asymptotically, both bounds given in Proposition 2 coincide yielding

$$\lim_{K=\zeta L\to\infty} \overline{C}_{\mathrm{ZFDF1}} = \log_2\left(1+\frac{1-\zeta}{\sigma^2}\right) - \frac{1+\sigma^2}{\zeta}\log_2\left(1-\frac{\zeta}{1+\sigma^2}\right) - \frac{1}{\ln(2)}. \tag{17}$$

Eq. (14) shows that the asymptotic average capacity is a strictly decreasing function of the number of users K. The number of users is upper bounded by the spreading factor, if zero–forcing equalization is addressed. This means, that the minimal asymptotic average capacity is achieved for $K = L$ and given by

$$\min_{\zeta \leq 1} \lim_{K=\zeta L \to \infty} \overline{C}_{\mathrm{ZFDF1}} = \left(1 + \sigma^2\right) \log_2\left(1 + \frac{1}{\sigma^2}\right) - \frac{1}{\ln(2)}. \tag{18}$$

Surprisingly, the asymptotic loss in capacity, i.e. the gap between orthogonal sequences and randomly chosen ones, is less than *one nat* (about 1.44 bits)

$$\min_{\zeta < 1} \lim_{K=\zeta L \to \infty} \log_2\left(1 + \frac{1}{\sigma^2}\right) - \overline{C}_{\mathrm{ZFDF1}} = \frac{1}{\ln(2)} - \sigma^2 \log_2\left(1 + \frac{1}{\sigma^2}\right). \tag{19}$$

Obviously, it has to vanishes for $\sigma^2 \to \infty$ and achieves the maximal value of one nat for $\sigma^2 \to 0$.

Comparing this with the results found for the optimum joint demodulator, see [3], we find that decorrelating decision–feedback equalization is asymptotically optimal if the number of users equals the spreading factor and the powers of the users are identical, but their rates differ. Moreover, it is the counterpart in multiuser equalization to a result found by Price [15, 16] for equalization of intersymbol interference where zero–forcing decision–feedback equalization is also asymptotically optimum as $\sigma^2 \to 0$.

Considering spectral efficiency instead of average capacity, a low number of users cannot be optimum any longer. With the help of (5) and (6), spectral efficiency is plotted as a function of power efficiency in Fig. 4 for $K \to \infty$. Note that the loss in spectral efficiency is larger than one nat for unit load. This is, as spectral efficiency is plotted versus the energy per *bit* which depends itself on the average capacity. In contrast, (19) is based on the energy per symbol. Nevertheless, (19) can be used to calculate the asymptotic loss in power efficiency. As the optimum load converges to 1 for large spectral efficiencies, cf. Figs. 4 or 5, the asymptotic power loss is bounded by

$$V_{\mathrm{ZFDF1}}(\Gamma) = \frac{\min\limits_{\zeta} \left(\dfrac{E_{\mathrm{b}}}{N_0}\right)_{\mathrm{ZFDF1}}}{\dfrac{2^{\Gamma} - 1}{\Gamma}} < \frac{\dfrac{2^{\Gamma + 1/\ln 2} - 1}{\Gamma + 1/\ln 2}}{\dfrac{2^{\Gamma} - 1}{\Gamma}} = \mathrm{e} + \mathcal{O}\left(\Gamma^{-1}\right). \tag{20}$$

As Ineq. (20) which becomes equality for $\Gamma \to \infty$ shows, the loss of only one nat in spectral efficiency — which obviously corresponds to a factor e in power efficiency — is achieved at least asymptotically for $\Gamma \to \infty$, i.e. $\frac{E_{\mathrm{b}}}{N_0} \to \infty$.

3.2 Equal Rates

In contrast to the equal power case discussed in the previous section, the preassumption of equal rates prohibits an explicit expression giving the average capacity, as the problem of optimal power assignment cannot be solved analytically for finite spreading gain, as (3) cannot be resolved with respect to C. Nevertheless, the average capacity can be bounded with the help of Proposition 1

$$\log_2\left(1 + \frac{L-k}{L} \cdot \frac{E_k}{N_0}\right) < \overline{C}_{\mathrm{ZFDF2}} < \log_2\left(1 + \frac{L-k+1}{L} \cdot \frac{E_k}{N_0}\right), \quad \forall 1 \leq k \leq K \tag{21}$$

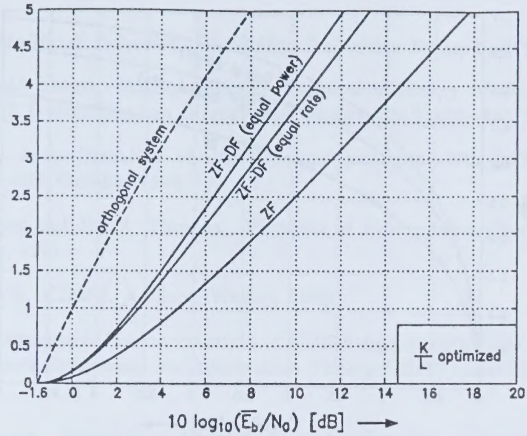


Figure 4: Power–bandwidth–plane for zero–forcing CDMA equalization with and without decision–feedback for optimized load. For comparison the dashed line shows an orthogonal system.

yielding bounds for the users' individual signal–to–noise ratios

$$\frac{2^{\overline{C}_{\mathrm{ZFDF2}}}-1}{1-(k-1)/L} < \frac{E_k}{N_0} < \frac{2^{\overline{C}_{\mathrm{ZFDF2}}}-1}{1-k/L}. \tag{22}$$

The energies per symbol and per bit averaged over all users are then bounded by

$$\frac{2^{\overline{C}_{\mathrm{ZFDF2}}}-1}{K}\sum_{k=1}^{K}\frac{L}{L-k+1} < \frac{\overline{E}_s}{N_0} < \frac{2^{\overline{C}_{\mathrm{ZFDF2}}}-1}{K}\sum_{k=1}^{K}\frac{L}{L-k} \tag{23}$$

and

$$\frac{2^{\Gamma L/K}-1}{\Gamma}\ln\left(\frac{L+1}{L-K+1}\right) < \left(\frac{\overline{E}_b}{N_0}\right)_{\mathrm{ZFDF2}} < \frac{2^{\Gamma L/K}-1}{\Gamma}\ln\left(\frac{L-1}{L-K-1}\right), \tag{24}$$

respectively. In the asymptotic case both bounds coincide. Thus, we have

$$\lim_{K=\zeta L\to\infty}\left(\frac{\overline{E}_b}{N_0}\right)_{\mathrm{ZFDF2}} = -\ln(1-\zeta)\,\frac{2^{\Gamma/\zeta}-1}{\Gamma}. \tag{25}$$

This relationship is illustrated for optimized load ζ in Fig. 4. The gap to the orthogonal bound is smaller than without decision–feedback, but slightly larger than in the equal power case. The asymptotic loss is bounded by

$$V_{\mathrm{ZFDF2}}(\Gamma) = \frac{\min\limits_{\zeta}\left(\frac{\overline{E}_b}{N_0}\right)_{\mathrm{ZFDF2}}}{\frac{2^{\Gamma}-1}{\Gamma}} = \frac{\inf\limits_{c}\lim\limits_{\zeta\to\frac{\Gamma}{\Gamma+c}}\left(\frac{\overline{E}_b}{N_0}\right)_{\mathrm{ZFDF2}}}{\frac{2^{\Gamma}-1}{\Gamma}} \tag{26}$$

for some positive constant c. With (25), the asymptotic loss can be shown by Taylor expansions to be

$$V_{\mathrm{ZFDF2}}(\Gamma) = \min_{c>0} 2^{c}\ln\left(1+\frac{\Gamma}{c}\right)\left(1+\mathcal{O}\left(2^{-\Gamma}\right)\right). \tag{27}$$

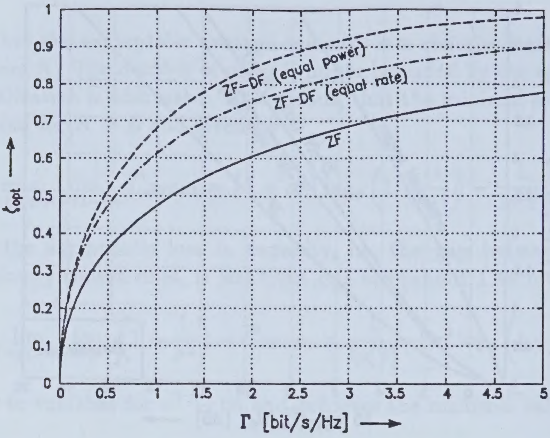


Figure 5: Ratios of users to spreading factor ζ_{opt} that maximize spectral efficiency Γ_{ZF} (solid line), Γ_{ZFDF1} (dashed line) and Γ_{ZFDF2} (dash–dotted line), respectively.

The asymptotic loss of the decorrelator is proportional to spectral efficiency, see (13), while it is constant for decorrelating decision–feedback with equal powers, cf. (20). For equal rates, it is shown in (27) to be proportional to the logarithm of spectral efficiency. This is a significant improvement compared to decorrelation without decision–feedback and only a small drawback to the equal power case.

4 Summary and Conclusions

Decorrelating multiuser detectors with and without decision–feedback have been analyzed. The performance variability of the decorrelating detector has been enabled to be derived analytically and the average capacity has been given for finite number of users. Moreover, the spectral efficiency has been shown to decrease slightly with the spreading factor, if the system load is fixed. Thus, it is *not* necessary to provide large spreading factors in order to neglect the degradation caused by performance variability. Decorrelating decision–feedback detectors have been found to achieve higher capacity when they are operated with equal power users rather than with equal rate users. This suggests the application of a time–sharing or rate–splitting [17] protocol to exploit all theoretical resources.

Acknowledgement

The author would like to thank Johannes Huber, Sergio Verdú, Peter Schramm and Alexander Lampe for stimulating hints and valuable discussions.

References

[1] Sergio Verdú. Minimum probability of error for asynchronous Gaussian multiple–access channels. *IEEE Transactions on Information Theory*, IT–32(1):85–96, January 1986.

[2] Sergio Verdú. *Multiuser Detection.* Cambridge University Press, New York, 1998.

[3] Sergio Verdú and Shlomo Shamai (Shitz). Spectral efficiency of CDMA with random spreading. *IEEE Transactions on Information Theory*, 45(2):622–640, March 1999.

[4] C. E. Shannon. A mathematical theory of communications. *The Bell System Technical Journal*, 27:623–656, October 1948.

[5] Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory.* John Wiley & Sons, New York, 1991.

[6] Andrew J. Viterbi. *CDMA.* Addison–Wesley, 1995.

[7] A.G. Burr. Performance of linear separation of CDMA signals with FEC coding. In *Proc. of IEEE International Symposium on Information Theory (ISIT)*, page 354, Ulm, Germany, June/July 1997.

[8] Branimir R. Vojcic. Information theoretic aspects of multiuser detection. In *Proc. of Symposium on Interference Rejection and Signal Separation for Wireless Communications (IRSS'97)*, Washington D.C., March 1997.

[9] David Tse and Stephen Hanly. Linear multiuser receivers: Effective interference, effective bandwidth and user capacity. *IEEE Transactions on Information Theory*, 45(2):641–657, March 1999.

[10] Upamanyu Madhow and Michael L. Honig. On the average near–far resistance for MMSE detection of direct sequence CDMA signals with random spreading. *To appear in IEEE Transactions on Information Theory*, 45, September 1999.

[11] Ralf R. Müller, Peter Schramm, and Johannes B. Huber. Spectral efficiency of CDMA systems with linear interference suppression (in German). In *Proc. of IEEE Workshop Kommunikationstechnik*, pages 93–97, Ulm, Germany, January 1997. ITUU–TR–1997/01. English version available via http://www-nt.e-technik.uni-erlangen.de/~dcg.

[12] Ralf R. Müller. Multiuser receivers for randomly spread signals: Fundamental limits with and without decision–feedback. *Submitted to IEEE Transactions on Information Theory*, June 1998.

[13] Ralf Müller. *Power and Bandwidth Efficiency of Multiuser Systems with Random Spreading.* Shaker–Verlag, Aachen, 1999.

[14] Ralf R. Müller and Johannes B. Huber. Capacity of cellular CDMA systems applying interference cancellation and channel coding. In *Proc. of Communication Theory Mini Conference (CTMC) at IEEE Globecom*, pages 179–184, Phoenix, AZ, November 1997.

[15] R. Price. Nonlinear feedback equalized PAM versus capacity for noisy filter channels. In *Proc. of IEEE International Conference on Communications (ICC)*, pages 22.12–22.17, Philadelphia, PA, June 1972.

[16] M. Vedat Eyuboğlu. Detection of coded modulation signals on linear severely distorted channels using decision–feedback noise prediction with interleaving. *IEEE Transactions on Communications*, 36(4):401–409, April 1988.

[17] Bixio Rimoldi and Rüdiger Urbanke. A rate–splitting approach to the Gaussian multiple–access channel. *IEEE Transactions on Information Theory*, 42(2):364–375, March 1996.

Scheduling of Switched Multibeam Antennas in a Multiple Access Environment*

Andrew Logothetis and H. Vincent Poor

Department of Electrical Engineering
Princeton University
Princeton, NJ 08544-5263

Abstract

Spatial diversity offered by antenna arrays for direct sequence code division multiple access (DS-CDMA) communication systems is exploited by designing an intelligent switched-beam antenna at the RF level. In this paper we address the design of optimal switched-beam antenna patterns, that yield conditional mean estimates of the communication channel during up-link transmission. The unknown transmitted signals received at the antenna elements are subject to multipath Rayleigh flat-fading. The aim is to compute minimum variance estimates of the communication channel by optimally combining the signals of the spatially distributed antennas at chip rate.

1 Introduction

The use of smart antennas is a very popular method designed to address the high demand in wireless communications when the limited radiofrequency spectrum is shared by an increasing number of users (See [1] and references therein for a complete, detailed discussions and issues concerning smart antennas). A smart antenna consists of an array of antenna elements and a beamformer which weighs and combines the outputs of the antenna elements. The aim of smart antennas is to increase the channel capacity by designing antenna beams that strengthen a desired user's signal, while at the same time attenuate the signals from other users/interferers. Systems employing smart antennas are divided into switch beam systems and adaptive antenna systems [2]:

In a switched-beam system, each cell site[1] is subdivided into a fixed number of narrow angular regions. Each region is serviced by one antenna beam. The beam yielding the strongest signal for a particular mobile user is selected for communication.

In an adaptive antenna system, a beam is formed in the direction of each mobile unit. The beam patterns are continuously updated, depending on the communication environment and the location of the mobile units.

A smart antenna is schematically shown in Fig. 1. The analog beamforming device takes as input the N antenna array outputs and generates according to the beamforming control, B (where $B < N$) antenna beams in the RF or IF level. The B output beams are then downconverted and sampled. Each beam is processed at baseband to yield multipath channel estimates. The estimated paths yielding the smallest estimation error are selected among all possible candidates. It should be noted that analog beamforming

---

*This research was supported in part by the U.S. Office of Naval Research under Grant N00014-94-1-0115, and in part by the New Jersey Center for Wireless Telecommunications

[1] A geographical region is subdivided into cells. At the center of each cell a fixed antenna array receives (or transmits) signals from (or to) a number of mobile units.

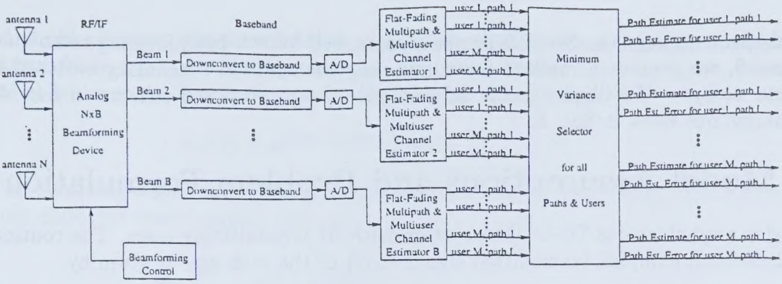

antenna 1
antenna 2
antenna N
RF/IF
Baseband
Analog
NxB
Beamforming
Device
Beam 1
Beam 2
Beam B
Downconvert to Baseband
A/D
Beamforming
Control
Flat-Fading
Multipath &
Multiuser
Channel
Estimator 1
Estimator 2
Estimator B
user 1, path 1
user 1, path L
user M, path 1
user M, path L
Minimum
Variance
Selector
for all
Paths & Users
Path Estimate for user 1, path 1
Path Est. Error for user 1, path 1
Path Estimate for user 1, path L
Path Est. Error for user 1, path L
Path Estimate for user M, path 1
Path Est. Error for user M, path 1
Path Estimate for user M, path L
Path Est. Error for user M, path L

Figure 1: A typical smart antenna, followed by an array of channel estimators. The smart antenna consists of an array of N antennas, a beamforming control and a beamforming device, which generates B antenna beams in the RF or IF level. The B output beams are downconverted and sampled. Each beam is processed at baseband to yield multipath channel estimates. The estimated paths yielding the smallest estimation error are selected among all possible candidates.

can be done at baseband, if the downconverting and sampling occurs immediately after the antenna outputs.

In this paper we design a new switched-beam antenna array receiver for direct sequence code division multiple access (DS-CDMA) communication systems operating in a multipath Rayleigh flat-fading channels. The switched-beam system operates at the RF or IF level. The novelty of the new switched-beam antenna array relays on designing a new beamforming device that switches between a candidate set of fixed beams *at chip rate*. A Kalman filter is used to estimate the attenuated spread spectrum signals for all users and all transmission paths. The schematics of the proposed switched-beam system is given by Fig. 2.

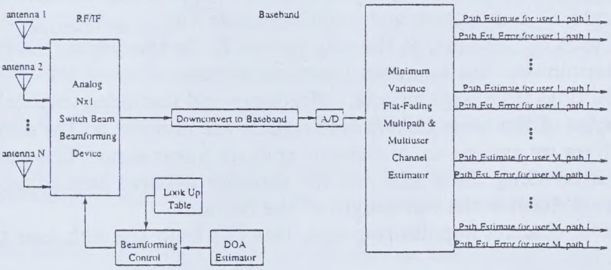

antenna 1
antenna 2
antenna N
RF/IF
Baseband
Analog
Nx1
Switch Beam
Beamforming
Device
Downconvert to Baseband
A/D
Look Up
Table
Beamforming
Control
DOA
Estimator
Minimum
Variance
Flat-Fading
Multipath &
Multiuser
Channel
Estimator
Path Estimate for user 1, path 1
Path Est. Error for user 1, path 1
Path Estimate for user 1, path L
Path Est. Error for user 1, path L
Path Estimate for user M, path 1
Path Est. Error for user M, path 1
Path Estimate for user M, path L
Path Est. Error for user M, path L

Figure 2: The proposed analog switched-beam antenna array receiver. A single beam is generated by the beamforming device at each time instant. The antenna beam pattern changes at chip rate according to a lookup table. The beam patterns and their switching sequence are designed to yield minimum variance estimates of the communication channel.

This paper is organized as follows: In Section 2 we describe the communication system, give the model assumptions and formulate our DS-CDMA switched-beam communication system as a linear, Gaussian, time-varying state-space model. In Sec. 2.1, we formulate the estimation objective, by seeking an optimal chip-rate beamforming scheduling (switching) sequence that yields minimum variance estimates of the received spread spectrum signals. Sec. 2.2 presents some well known beamforming techniques. In Section. 3, we present a method for computing the optimal scheduling switched-beam antenna arrays. We illustrate the advantages of our proposed scheme in Sec. 4 and summarize our work in Sec. 5.

2 Model Assumptions and Problem Formulation

Consider a synchronous DS-CDMA system with M transmitting users. The continuous-time baseband complex transmitted signal $s_m(t)$ of the mth user is given by

$$s_m(t) \;=\; \sqrt{2P_m}b_m(t)c_m(t)e^{j\phi_m} \tag{1}$$

where $P_m, b_m(t), c_m(t)$ and ϕ_m, denote the transmitted power, data symbol, the spreading signal, and phase of the mth user. The spreading signal $c_m(t)$ is given by

$$c_m(t) = \sum_{k=-\infty}^{\infty} c_{k,m}\mathrm{rect}_{T_c}(t-kT_c) \tag{2}$$

where $\mathrm{rect}_{T_c}(t)$ is a unit rectangular pulse of duration T_c, and $c_{k,m} \in \{-1,1\}$ denotes the kth chip of the mth user. The spreading code sequences $\{c_{k,m}\}$ for all users are known to the receiver.

Let T_b denote the symbol (or bit) duration. The ratio $G = T_b/T_c$ is known as the processing gain. In this paper we assume G to be an integer. The data signal $b_m(t)$ is given by

$$b_m(t) = \sum_{n=-\infty}^{\infty} b_{n,m}\mathrm{rect}_{T_b}(t-nT_b) = \sum_{k=-\infty}^{\infty} b^c_{k,m}\mathrm{rect}_{T_c}(t-kT_c) \tag{3}$$

where $b_{n,m} \in \{-1,1\}$ is the nth data bit of the mth user. Let $b_{n,m} = b^c_{0+nG,m} = b^c_{1+nG,m} = \cdots = b^c_{G-1+nG,m}$ for $n \in \{-\infty,\ldots,\infty\}, m \in \{1,\ldots,M\}$. Thus, we have subdivided the symbol $b_{n,m}$ into G consecutive and equal symbols $\{b^c_{0+nG,m}, b^c_{1+nG,m},\cdots,b^c_{G-1+nG,m}\}$. The period of each $b^c_{k,m}$ is equal to the chip period T_c. In this paper, the data signals are treated as deterministic, but unknown functions of time.

Due to reflections from large objects, diffractions and scattering, multiple, delayed and attenuated copies of the same transmitted signals are induced on the different elements of the array. Here we assume an N element uniform linear array. The antenna elements are equally spaced along the x axis and the distance between two adjacent elements is $d = \lambda/2$, where λ denotes the wavelength of the carrier.

The N vector channel impulse response, (see [2]) from the mth user to the antenna array is given by

$$h_m(t,\tau) = \sum_{l=1}^{L_m(t)} a(\phi_{m,l})\rho_{m,l}(t)e^{j\psi_{m,l}(t)}\delta(t-\tau_{m,l}(t)) \tag{4}$$

where $L_m(t)$ is the known - possibly time varying - number of resolvable paths. $\delta(t)$ is the Dirac delta function. $\phi_{m,l}(t)$ is the known direction of arrival (DOA) or angle of arrival (AOA) of the mth user's signal via the lth path. The AOA is measured anti-clockwise from positive real axis. $\rho_{m,l}(t)$, $\psi_{m,l}(t)$ and $\tau_{k,l}(t)$ is the signal attenuation, phase shift and time delay of the lth multipath component for user m. $a(\phi)$ is known as the steering vector, and denotes the response of the antenna array to a signal impinging onto the array from an angle ϕ. For a uniform linear array, the steering vector is given by

$$a(\phi) = [\; g_1(\phi)a_1(\phi),\;\; g_2(\phi)a_2(\phi),\;\; \ldots,\;\; g_N(\phi)a_N(\phi)\;] \tag{5}$$

where $g_n(\phi)$ denotes the directivity (gain) of the nth antenna element impinging on the array from angle ϕ. Here we assume that all the elements have unit directivity, i.e. $g_1(\phi) = \cdots = g_N(\phi) = 1$. For our uniform linear array we have

$$a_n(\phi) = e^{j\pi(n-1)\cos(\phi)}, \ \forall n \in \{1, \ldots, N\} \tag{6}$$

The receiving signal at the array elements is given by the superposition of all the channel distorted users' signals, that is

$$\begin{aligned} r(t) &= \sum_{m=1}^{M} s_m(t) * h_m(t, \tau) \\ &= \sum_{m=1}^{M} \sum_{l=1}^{L_m(t)} a(\phi_{m,l})\alpha_{m,l}(t)b_m(t - \tau_{m,l}(t))c_m(t - \tau_{m,l}(t)) \end{aligned} \tag{7}$$

where $*$ denotes the convolution operator, and the complex amplitude $\alpha_{m,l}(t)$

$$\alpha_{m,l}(t) = \sqrt{2P_m\rho_{m,l}(t)}e^{j(\psi_{m,l}(t)+\phi_m)} \tag{8}$$

denotes the attenuated, phase shifted multiplicative channel interference at time t via the lth path for the m user.

Prior to sampling, the signals from each element of the antenna array are multiplied with a complex number and then summed to form the array output. The process of combining the signals at the array is known as beamforming. The output of the beamforming array is given by

$$y(t) = f^H(t)r(t) + f^H(t)w(t) \tag{9}$$

where $w(t)$ is the vector representing the additive background noise, $(\cdot)^H$ denotes the complex transpose operator, and $f(t)$ denotes the time-varying beamforming vector.

A discrete time version of Eq. (9) is given by sampling the output of the beamformer at fixed intervals. In this paper, sampling rate is equal to the chip rate. The scalar complex discrete-time beamformer output is thus given by

$$y(kT_c + \Delta) = f^H(kT_c + \Delta)r(kT_c + \Delta) + f^H(kT_c + \Delta)w(kT_c + \Delta) \tag{10}$$

for $k = \ldots, -2, -1, 0, 1, 2, \ldots$ The sampling offset is denoted by Δ, such that $0 \leq \Delta < T_c$.

Model and Statistical Assumptions: We now present some important assumptions for our analysis.

Flat fading: We model the communication channel as flat-fading. The delay spread, which is defined as the inverse of the channel coherence bandwidth [3], is assumed much smaller than the symbol period. Here, we further assume that, $0 \leq \tau_{m,l}(t) < T_c$, for all time t, users m, and propagation paths l. Let the maximum number of paths $L_m(t)$ for each user to be equal to L.

Sampling: We chose our sampling instances, such that the sampling offset Δ satisfies the following condition

$$0 \leq \tau_{m,l}(t) < \Delta < T_c, \forall t, m \in \{1, \ldots, M\}, l \in \{1, \ldots, L\} \tag{11}$$

As a consequence of Eq. (11), the users appear synchronous at the receiver.

Independent Rayleigh distributed multiplicative channel coefficients: We assume the channel coefficients $\alpha_{m,l}(t)$ in Eq. (8) between symbol intervals are modeled as independent zero-mean circular white Gaussian noise processes and remain constant during each symbol period. In particular, for $n, n' = -\infty, \ldots, \infty$, $m, m' \in \{1, \ldots, M\}$ and $l, l' \in \{1, \ldots, L\}$,

$$\alpha_{m,l}(nT_b + \Delta) \sim N(0, \sigma^2_{m,l}) \tag{12}$$

$$\alpha_{m,l}(nT_b + \Delta) = \alpha_{m,l}(nT_b + T_c + \Delta) = \cdots \alpha_{m,l}(nT_b + (G-1)T_c + \Delta) \tag{13}$$

$$E\{\alpha_{m,l}(nT_b + \Delta)\alpha_{m',l'}(n'T_b + \Delta)\} = \delta_{n,n'}\delta_{m,m'}\delta_{l,l'}\sigma^2_{m,l} \tag{14}$$

where $\delta_{k,k'}$ denotes the Kronecker delta function, that is $\delta_{k,k'} = 1$, if $k = k'$, and $\delta_{k,k'} = 0$ otherwise.

Temporally white observation noise: The noise vector $w(kT_c + \Delta)$, is modeled as a zero-mean circular white Gaussian noise process with a known $N \times N$ covariance matrix R. For $k = -\infty, \ldots, \infty$, we have

$$\begin{aligned} E\{w(kT_c + \Delta)\} &= 0_{N\times 1} \\ E\{w(kT_c + \Delta)w^H(k'T_c + \Delta)\} &= R\delta_{k,k'} \end{aligned} \tag{15}$$

where $0_{N\times 1}$ denotes a $N \times 1$ vector of zeros.

Discrete-time state-space model. For notational convenience, we rewrite the kth sampled array output after beamforming, given by Eq. (10), as follows

$$y_k = f_k^H r_k + f_k^H w_k \tag{16}$$

where $y_k \triangleq y(kT_c + \Delta)$, $f_k \triangleq f(kT_c + \Delta)$ and $w_k \triangleq w(kT_c + \Delta)$. Here we use the subscript k to denote the the kth sample.

Let $x_{k,m,l}$ denote the the attenuated received spread spectrum signal at time k for the mth user via the lth path, i.e

$$x_{k,m,l} \triangleq b^c_{k,m} c_{k,m} \alpha_{k,m,l} \tag{17}$$

where $\alpha_{k,m,l} \triangleq \alpha_{m,k}(kT_c + \Delta)$. Let

$$v_{k,m,l} = \begin{cases} x_{k,m,l}, & \forall k \in \{\ldots, -2(G-1), -(G-1), 0, G-1, 2(G-1), \ldots\} \\ 0, & \text{otherwise} \end{cases} \tag{18}$$

denote the realization of the attenuated and phase shifted spread spectrum $x_{k,m,l}$ every G samples. Since the product $b^c_{k,m} c_{k,m}$ yields ± 1, and from our statistical assumptions on the channel fading, $v_{k,m,l}$ is complex white Gaussian distributed random variable, such that

$$v_{k,m,l} \sim N(0, \sigma^2_{m,l}), \text{ for } k \in \{\ldots, -2(G-1), -(G-1), 0, G-1, 2(G-1), \ldots\} \tag{19}$$

and $v_{k,m,l} = 0$, otherwise. Let

$$h_k = \begin{cases} 0, & \forall k \in \{\ldots, -2(G-1), -(G-1), 0, G-1, 2(G-1), \ldots\} \\ 1, & \text{otherwise} \end{cases} \tag{20}$$

The attenuated phase shifted received spread spectrum signal $x_{k,m,l}$ can be written in the following scalar recursive form

$$x_{k,m,l} = h_k c_{k,m} c_{k-1,m} x_{k-1,m,l} + v_{k,m,l} \tag{21}$$

We group the received signals $x_{k,m,l}$ at time index k in a $ML \times 1$ vector x_k as follows

$$x_k^H \triangleq (x_{k,1,1}, \ldots, x_{k,1,L}, \cdots, x_{k,M,1}, \ldots, x_{k,M,L})^H \tag{22}$$

If we define $v_k^H \triangleq (v_{k,1,1}, \ldots, v_{k,1,L}, \cdots, v_{k,M,1}, \ldots, v_{k,M,L})^H$, then we have the following vector state equation for the time evolution of the received spread spectrum signals for all user and all paths

$$x_k = A_k x_{k-1} + v_k \tag{23}$$

where

$$\begin{aligned} v_k &\sim N(0_{ML\times 1}, Q), \forall\, k \in \{\ldots, -2(G-1), -(G-1), 0, G-1, 2(G-1), \ldots\} && (24)\\ v_k &= 0, \text{otherwise} && (25)\end{aligned}$$

and

$$\begin{aligned} A_k &= \operatorname{diag}\left(\underbrace{h_k c_{k-1,1} c_{k,1}, \ldots, h_k c_{k-1,1} c_{k,1}}_{L \text{ times}}, \ldots, \underbrace{h_k c_{k-1,M} c_{k,M}, \ldots, h_k c_{k-1,M} c_{k,M}}_{L \text{ times}} \right) && (26)\\ Q &= \operatorname{diag}\left(\sigma_{1,1}^2, \ldots, \sigma_{1,L}^2, \cdots, \sigma_{M,1}^2, \ldots, \sigma_{M,L}^2\right) && (27)\end{aligned}$$

where A_k and Q are $ML \times ML$ diagonal matrices.

Let ϕ denote the vector of all angle of arrivals for all M users

$$\phi^H = (\phi_{1,1}, \ldots, \phi_{1,L}, \cdots, \phi_{M,1}, \ldots, \phi_{M,L})^H \tag{28}$$

We use $A(\phi)$ to denote the $N \times ML$ matrix response of the array, defined as follows

$$A(\phi) = (a(\phi_{1,1}), \ldots, a(\phi_{1,L}), \cdots, a(\phi_{M,1}), \ldots, a(\phi_{M,L})) \tag{29}$$

The $((m-1) \times L + l)$th column vector in $A(\phi)$ corresponds to the antenna array vector response of a signal impinging onto the antenna array from angle $\phi_{m,l}$.

Using Eqs. (23) and (29), the observation Eq. (16) is equivalently written as

$$y_k = f_k^H A(\phi) x_k + f_k^H w_k \tag{30}$$

Equations (23) and (30) form the discrete-time, state-space version of the DS-CDMA beamforming linear antenna array system.

2.1 Optimal Chip Rate Switched-Beam Design

In this subsection we formulate the estimation objectives. The aim is to design the time-varying switched-beam beamforming vector f_k, such that the minimum estimation error in estimating the received spread spectrum signal x_k defined in Eq. (22) is obtained. We chose the beamforming vector f_k to be a function of the data from time index 0 up to time index $k-1$. Furthermore, f_k belongs to a set of fixed-beam array patterns F. Thus,

$$f_k = f_k(y_0, y_1, \ldots, y_{k-1}) \in F \tag{31}$$

for $k = 1, 2, \ldots$. The aim is to select the class of beam patterns F and optimally select the sequence of beam-patterns from the set F, such that the estimation errors in estimating the spread spectrum signals are minimized. We chose the following optimization function

$$h_k = E\{(x_k - x_{k|k})^H J\,(x_k - x_{k|k})\} \tag{32}$$

where $x_{k|k} \triangleq E\{x_k | Y_k\}$ is the conditional mean estimate of x_k, conditioned on the data $Y_k \triangleq (y_0, y_1, \ldots, y_k)$. J is some predetermined - user defined - weighting matrix. We chose the sequence of optimal beamforming vector $(\hat{f}_0, \hat{f}_1, \hat{f}_2, \ldots)$, as follows

$$(\hat{f}_0, \hat{f}_1,, \ldots, \hat{f}_k) = \underset{f_0, f_1,, \ldots, f_k}{\operatorname{argmax}}\, E\left\{(x_k - x_{k|k})^H J\,(x_k - x_{k|k})\right\} \tag{33}$$

subject to Eq. (31)

In the following subsection we present several well known beamforming techniques.

2.2 Beamforming Techniques

We group our beamforming techniques according to

- Fixed-beam system, where the antenna beams are fixed in azimuth.
- Maximizing the response of the antenna array in the direction of a mobile unit.
- Minimizing the average output power of the beamforming array, while ensuring unity response in a desired direction.

We now present details on selecting the sets of beamforming array patterns based on the three beamforming techniques mentioned above.

Fixed-Beam System The beamforming vectors in a fixed-beam antenna array systems, are designed to jointly span the entire area around the cell site. Let B denote the number of fixed beams and ϕ_i denote the ith angle of the beam where the antenna array is pointing. The set of fixed-beams are given by

$$F_1 = \{a(\phi_1), a(\phi_2), \cdots, a(\phi_B)\} \tag{34}$$

Conventional Beamformer The beamforming vector in a conventional beamformer are proportional to the steering vector in the known direction of the desired signal [4]. The complex beamforming vector weights have equal magnitudes and the phases are set to the phases of the array vector response when the angle of arrival is in the desired direction θ.

From previous discussion we assumed $M \times L$ steering vectors associated with $M \times L$ known angle-of-arrivals $\phi_{m,l}$. The set of conventional beamforming array vectors are given by

$$F_2 = \{a(\phi_{1,1}), \ldots, a(\phi_{1,L}), \cdots, a(\phi_{M,1}), \ldots, a(\phi_{M,L})\} \tag{35}$$

Optimal Beamforming The optimal beamforming vector is derived by minimizing the average output power of the beamforming array, while ensuring the unity response in the desired direction s_0 [4]. Assuming $f = f_k$ for all k, we solve the following optimization problem

$$\min_f E\{yy^H\}, \text{ subject to } f^H s_0 = 1 \tag{36}$$

Using Langrange's multiplier method, the optimal beamformer f can be expressed as follows

$$f = \frac{D^{-1}s_0}{s_0^H D^{-1} s_0} \tag{37}$$

where D denotes the average signal, multi-user interference (MUI) plus noise power. From our statistical assumptions on our signal model, we can show the following result

$$D = A(\phi)QA^H(\phi) + R \tag{38}$$

Note the D is positive definite matrix since Q and R are positive definite matrices. Thus, the set of optimal beamforming steering vectors are given by

$$\begin{aligned} F_3 = & \left\{ \frac{D^{-1}a(\phi_{1,1})}{a^H(\phi_{1,1})D^{-1}a(\phi_{1,1})}, \ldots, \frac{D^{-1}a(\phi_{1,L})}{a^H(\phi_{1,L})D^{-1}a(\phi_{1,L})}, \cdots, \frac{D^{-1}a(\phi_{M,1})}{a^H(\phi_{M,1})D^{-1}a(\phi_{M,1})}, \ldots, \right. \\ & \left. \frac{D^{-1}a(\phi_{M,L})}{a^H(\phi_{M,L})D^{-1}a(\phi_{M,L})} \right\} \end{aligned} \tag{39}$$

3 Optimal Scheduling of Switched Multibeam Antennas

We now summarize our state-space model, estimation objectives and present the optimal switched-beam sequence f_k, in the following theorem:

Theorem 3.1 *Given*
System model: The multipath Rayleigh fading DS-CDMA spread spectrum communication system sampled at chip rate at the antenna array output after beamforming is described by the following discrete time equation

$$\begin{aligned} x_{k+1} &= A_{k+1} x_k + v_{k+1} & (40)\\ y_k &= f_k^H A(\phi) x_t + f_k^H w_t, \quad \text{for } f_k \in F & (41) \end{aligned}$$

where k is the time index, x_k is the state vector representing the received attenuated and phase shifted versions of the DS-CDMA signals. The state transition matrix A_k, is given by Eq. (26) and the process noise v_t is a zero-mean complex white Gaussian noise sequence given by Eqs (24) and (25). f_k is the beamforming vector and it is an element of a finite set F of preselected steering vectors. w_k is the noise vector at the antenna array, and w_k is modeled as a zero-mean complex white Gaussian measurement noise sequence with positive definite covariance R. Finally, we assume that that the process noise v_k and observation noise w_k are statistically uncorrelated, that is $E\{w_k v_l^H\} = 0$ for $k, l = 0, 1, 2, \ldots$.
Known Model Parameters: The angle-of-arrivals ϕ for all spread spectrum users and their associated steering vectors are known, thus $A(\phi)$ is assumed known. The spreading sequence are also know to the receiving antenna array, thus A_k are known for all k. Finally, the noise covariance matrices Q and R are also known. When some of the model parameters are not known, then the unknown parameters can be estimated. Model parameter estimation is not investigated here.
Constraints: We chose F to be equal to one of the three beamforming vector sets discussed in Section 2.2. Only one beamforming vector $f_k \in F$ may be used at any one time. The active beamforming vector at time k is selected via one of the following control laws

$$\begin{aligned} \tilde{f}_k &= \tilde{f}(k, A(\phi), A_k, Q, R) \in F & (42)\\ f_k &= f(k, A(\phi), A_k, Q, R, Y_{k-1}) \in F & (43) \end{aligned}$$

The *closed-loop control function* $f(k, A(\phi), A_k, Q, R, Y_{k-1})$ - to be determined - at time k makes use of all available information up to time k. Y_k denotes the measurement sequence $(y_0, y_1, \ldots, y_k)$. Note that f_k is a function of the data Y_{k-1} and thus the measurement y_k is also function of Y_{k-1}. The *open-loop control function* $\tilde{f}(k, A(\phi), A_k, Q, R)$ at time k is a function of the known model parameters and a function of time.
Objective functions: Compute the chip-rate beamforming vector sequences $\widehat{F}_{G-1} \triangleq (\widehat{f}_0, \widehat{f}_2, \ldots, \widehat{f}_{G-1})$ and $\widehat{\tilde{F}}_{G-1} \triangleq (\widehat{\tilde{f}}_0, \widehat{\tilde{f}}_1, \ldots, \widehat{\tilde{f}}_{G-1})$ which respectively maximize the terminal cost function $h_{G-1} = E\left\{(x_{G-1} - x_{G-1|G-1})^H J\,(x_{G-1} - x_{G-1|G-1})\right\}$ as follows

$$\begin{aligned} \widehat{\tilde{F}}_{G-1} &= \underset{\tilde{F}_{G-1}}{\operatorname{argmax}} E\left\{(x_{G-1} - x_{G-1|G-1})^H J\,(x_{G-1} - x_{G-1|G-1})\right\} & (44)\\ \widehat{F}_{G-1} &= \underset{F_{G-1}}{\operatorname{argmax}} E\left\{(x_{G-1} - x_{G-1|G-1})^H J\,(x_{G-1} - x_{G-1|G-1})\right\} & (45) \end{aligned}$$

Then the optimal control sequences $(\widehat{f}_0,\ldots,\widehat{f}_{G-1})$ *and* $(\widehat{\widetilde{f}}_0,\ldots,\widehat{\widetilde{f}}_{G-1})$ *obtained from (44) and (45) respectively, are computed off-line by solving a nonlinear deterministic optimization problem. In particular,*

1. $\widehat{f}_k = \widehat{\widetilde{f}}_k$, *for* $k = 0, 1, \ldots, G-1$
2. $\widehat{F}_{G-1} = \operatorname{argmax}_{F_{G-1}} \operatorname{Tr}\left\{J\, P_{G-1|G-1}(F_{G-1})\right\}$

where $P_{G-1|G-1}(F_{G-1}) \triangleq E\left\{(x_{G-1} - x_{G-1|G-1})(x_{G-1} - x_{G-1|G-1})^H | Y_{G-1}, F_{G-1}\right\}$ *denotes the error covariance matrix of* x_{G-1} *given the data* Y_{G-1}. $P_{G-1|G-1}(F_{G-1})$ *is the solution of the discrete-time Riccati equation, computed by running a Kalman filter [5], as follows*

$$
\begin{aligned}
P_{k|k-1}(F_{k-1}) &= A_k P_{k-1|k-1}(F_{k-1}) A_k^H + Q && (46)\\
S_{k|k-1}(F_k) &= f_k^H A(\phi) P_{k|k-1}(F_{k-1}) A^H(\phi) f_k + f_k R f_k^H && (47)\\
P_{k|k}(F_k) &= P_{k|k-1}(F_{k-1})\left[I - f_k A^H(\phi) S_{k|k-1}^{-1}(F_k) f_k^H(\phi) P_{k|k-1}(F_{k-1})\right] && (48)
\end{aligned}
$$

for $k = 0, \ldots, G-1$ *and* $P_{-1|-1}(F_{-1}) = 0_{ML \times ML}$.

PROOF. The proof follows similarly to the proof in [6], where dynamic programming is used to compute the optimal control for controlling the measurement equations in a Linear Quadratic Control (LQG) problem. □

Thus, the error covariance matrix $P_{G-1|G-1}(F_{G-1})$, as shown from the Riccati equations above, is data independent. Furthermore, the open-loop and closed-loop beamforming vector sequence are identical. The optimal sequence is derived by using either: 1) dynamic programming, 2) enumeration and 3) enumeration via optimal pruning [7].

Dynamic programming is not suitable for this particular application, since the dynamic programming solution requires the quantization of a large dimensional error covariance matrix. The optimal beamforming sequence via enumeration is computed by considering all possible beamforming vector sequences and choosing the sequence that yields the minimum terminal cost. In [7], an algorithm that reduces the complexity in searching the optimal sequence is derived. The algorithm in [7] is an enumeration scheme with optimal pruning where sets of sequences are delete without the possibility of removing the optimal sequence.

4 An Illustrative Example

We compare our proposed optimal scheduling switch multibeam antenna with the non-scheduled multibeam antenna configuration of Fig.1. We investigate the errors in estimating the channel coefficients using a linear antenna array consisting of three elements. The number of spread spectrum users and transmission (multi) paths are both set to two. The processing gain is set to $G = 6$. Our results are averaged over 200 Monte Carlo simulations. On each simulation we randomly vary the DOA of the received signals, the channel fading, and the PN code of each transmitting user. During our simulation studies we keep the variance of the observation noise and the Rayleigh fading fixed at, 0.001 and 1, respectively. In Fig. 4, we show the improvement in optimally scheduling multibeam antenna systems. Note also, that the fixed switch-beam antenna yield the best result.

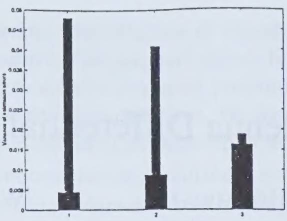


Figure 3: Variance of estimation errors, averaged over all paths. Wide bars denote optimal scheduling. Narrow bars denote non-switched beamforming. The first, second and third column, respectively correspond to Fixed, Conventional, and Optimal Beamforming.

5 Conclusion

In this paper, we considered the problem of optimally scheduling a multibeam antenna array for DS-CDMA communication systems operating in a multipath Rayleigh flat-fading channel. We formulated the received communication system as a linear, Gaussian, time-varying state-space model. We derived a scheme that yields the minimum estimates of the received spread spectrum signals, by computing optimal chip-rate beamforming scheduling (switching) sequences.

The advantages of our proposed scheme are: 1) Only one beam is generated at each time instant. The generated antenna beam is used to estimate all spread spectrum users and their transmission paths. 2) The beam pattern of the antenna array is selected among a predetermined set of beam patterns. 3) The beam pattern changes at chip rate. 4) The switching sequence of the antenna beams is *data independent* when the direction of arrival (DOA) of the received spread spectrum signals are known. Thus, the switching sequence is determined *a priori*. 5) The optimal switched beam sequence is designed to yield minimum estimates of the communication channel.

References

[1] *Smart Antennas: Adaptive Arrays, Algorithms, & Wireless Position Location*, Edited by T.S. Rappaport, IEEE, 1998.

[2] J.C. Liberti and T.S. Rappaport, *Smart Antennas for Wireless Communications: IS-95 and Third Generation CDMA Applications*, Prentice Hall, 1999.

[3] L.C. Godara, *Applications of Arrays to Mobile Communications, Part I: Performance Improvements, Feasibility, and System Considerations*, IEEE Proceedings, Vol. 85, No. 7, July 1997, pp. 1031-1060.

[4] L.C. Godara, *Applications of Arrays to Mobile Communications, Part II: Beam-Forming and Direction-Of-Arrival Considerations*, IEEE Proceedings, Vol. 85, No. 8, August 1997, pp. 1195-1245.

[5] B.D.O. Anderson and J.B. Moore. *Optimal Filtering*, Englewood Cliffs, NJ, Prentice-Hall 1979.

[6] L. Meier, J. Peschon and R. M. Dressler, *Optimal Control of Measurement Subsystems*, IEEE Transactions on Automatic Control, Vol. 12, No. 5, pp. 528-536, October 1967.

[7] A. Logothetis and A. Issakson, *On Sensor Scheduling via Information Theoretic Criteria*, American Control Conference, 1999.

Multiple Antenna Differential Modulation

BERTRAND M. HOCHWALD WIM SWELDENS

Bell Laboratories, Lucent Technologies
600 Mountain Avenue, Murray Hill, NJ 07974
{hochwald,wim}@bell-labs.com

September 21, 1999

Abstract

We present a framework for differential modulation with multiple antennas across a continuously fading channel, where neither the transmitter nor the receiver knows the fading coefficients. The framework can be seen as a natural extension of standard differential phase shift keying (DPSK) commonly used in single-antenna unknown-channel systems. We show how our differential framework links the unknown-channel system with a known-channel system. As a special case, we introduce a class of diagonal signals where only one antenna is active at any time, and show how these signals may be used to achieve full transmitter diversity and low probability of error.

1 Introduction

Recent advances in communicating across multiple-antenna wireless communication links show that these links can support very high data rates with low error probabilities, especially when the wireless channel response is known at the receiver [1, 2]. However, the assumption that the channel is known is questionable in a rapidly changing mobile environment, or when multiple transmitter antennas are employed. In [3], a new class of signals called *unitary space-time* signals is proposed that is well-tailored for Rayleigh flat-fading channels where neither the transmitter nor the receiver knows the fading coefficients. The unitary space-time signals are suited particularly well to piecewise-constant fading models. In this note, we show how to modify these signals to work when the fading changes continuously. The modified signals, which we denote *differential unitary space-time modulation*, are easily implemented and achieve full-antenna diversity.

Differential phase-shift keying (DPSK) has long been used in single-antenna unknown-channel links when the channel has a phase response that is approximately constant from one time sample to the next. Differential modulation encodes the transmitted information into phase differences from symbol to symbol. The receiver decodes the information in the current symbol by comparing its phase to the phase of the previous symbol. DPSK is widely used because many continuously fading channels change little between successive time samples.

We introduce a generalization of DPSK to $M > 1$ transmitter antennas. We encode information in blocks of M symbols and transmit $M \times M$ matrices where each row contains the symbols transmitted across the M antennas at each time sample. Any two successive blocks contain information embedded in their quotient.

We consider particularly simple constellations of signals that are *diagonal*, where at any given time only one antenna is active. The diagonal signals fully utilize the transmitter antenna diversity and can be optimized to achieve low error probability across a Rayleigh flat-fading channel. Several examples and simulations are given. We show that our differential framework provides a connection to multiple-antenna communication when the channel is known [4, 5].

Throughout the paper, we concentrate on modulation, demodulation, and constellation design issues, and do not address coding issues that lower error probability by adding redundancy. We focus, instead, on raw or uncoded signal and bit error probabilities.

Note: This paper is an abridged version of the paper "Differential Unitary Space-Time Modulation" [6] which can be downloaded in its entirety from from

`http://cm.bell-labs.com/who/wim/papers/dust/`

2 Single Antenna Differential Modulation

In this section we give a short review of single-antenna differential phase shift keying (DPSK); this review sets the stage for the multiple-antenna differential modulation technique we discuss in the next section.

In the channel model, the received signals are corrupted by additive noise and fading. We use complex baseband notation: at time t we transmit the signal s_t and we receive the noisy signal x_t at the receiver antenna. The action of the channel is given by:

$$x_t = \sqrt{\rho}\, h_t\, s_t + w_t, \qquad t = 0, 1, \ldots. \tag{1}$$

The additive noise at time t denoted w_t, and is independent, identically distributed $\mathcal{CN}(0,1)$ which is the complex Gaussian zero-mean unit-variance distribution. Here h_t is the complex-valued fading coefficient between the transmitter receiver antenna. The fading coefficients are $\mathcal{CN}(0,1)$ distributed but not independent. We assume that the fading coefficients change continuously according to a model such as Jakes' [7]. Assuming that the signals have unit modulus, ρ represents the expected signal-to-noise ratio (SNR) at the receiver.

For a data rate of R bits per channel use, we need $L = 2^R$ symbols. A common technique is phase shift keying (PSK) which uses symbols that are Lth roots of unity: $v_\ell = e^{2\pi i \ell/L}$ with $\ell = 0, \ldots, L-1$. Suppose we want to send a data sequence of integers $z_1, z_2, \ldots$ with $z_t \in \{0, \ldots, L-1\}$. The transmitter sends the symbol stream $s_1, s_2, \ldots$ where

$$s_t = v_{z_t}\, s_{t-1} \qquad t = 1, 2, \ldots \quad (s_0 = 1). \tag{2}$$

If we assume that the fading coefficient changes slowly so that $h_t \approx h_{t-1}$, then the maximum likelihood decoder is given by

$$\hat{z}_t = \arg\min_l |x_t - v_l\, x_{t-1}|^2 = \lfloor \arg(x_t/x_{t-1})\, L/(2\pi) \rceil,$$

where $\lfloor x \rceil$ stands for the integer closest to x.

3 Multiple Antenna Differential Modulation

Consider a communication link comprising M transmitter antennas and N receiver antennas that operates in a Rayleigh flat-fading environment. Each receiver antenna responds to each transmitter antenna through a statistically independent fading coefficient. The received signals are corrupted by additive noise that is statistically independent among the N receivers and the symbol periods. We assume that the fading coefficients are constant over blocks of length M. The action of the channel can then be written as:

$$X_\tau = \sqrt{\rho}\, S_\tau\, H_\tau + W_\tau \qquad \tau = 0, 1, \ldots \tag{3}$$

Here X_τ is a $M \times N$ matrix of received signals and S_τ is $M \times M$ matrix of transmitted signals. In this notation, the M columns of S_τ represent the signals sent on the M transmitter antennas as functions of time. H_τ is an $M \times N$ matrix of complex-valued fading coefficient between the transmitter receiver antennas. The fading coefficients are independent across antenna but not across time. The additive noise is a $M \times N$ matrix of independent identically $\mathcal{CN}(0, 1)$ distributed coefficients. The fading coefficients again change continuously according to a model such as Jakes' [7]. The expected combined energy of the signals sent on the different antenna at each time step is normalized to be one. This way the total transmitted power does not depend on the number of transmitter antennas and ρ represents the expected signal-to-noise ratio (SNR) at each receiver antenna.

When the fading coefficients are known to the receiver the maximum likelihood decoder is given by

$$\hat{z}_\tau = \arg\min_{\ell} \|X_\tau - \sqrt{\rho} V_\ell H_\tau\|\,, \tag{4}$$

where the norm is the Frobenius norm. However, in a mobile setting one can not necessarily assume that sufficient time or resources are available for the receiver to learn the channel. Thus, we do not assume that the receiver knows H_τ.

The generalization of differential modulation to multiple antennas is presented in detail in [6] and we review the main ideas. We satisfy the power constraint by using unitary matrices S_τ. Given that a single block takes up M uses of the channel, a rate of R bits/channel use requires $L = 2^{RM}$ different signal matrices. Each signal is an $M \times M$ unitary matrix V_ℓ from a constellation $\mathcal{V}$ of L such matrices. The bits to be transmitted are represented by an integer data sequence $z_1, z_2, \ldots$ with $z_\tau \in \{0, \ldots, L-1\}$. The fundamental differential encoding equation is given by

$$S_\tau = V_{z_\tau}\, S_{\tau-1} \qquad \tau = 1, 2, \ldots. \tag{5}$$

Clearly, all transmitted matrices $\mathcal{S}_\tau$ will be unitary matrices. Note the formal similarity between (5) and (2).

We assume that the fading coefficient are approximately constant over $2M$ time samples ($H_{\tau-1} \approx H_\tau$). We can now substitute the above equation in (3) to obtain

$$X_\tau = V_{z_\tau}\, X_{\tau-1} + W_\tau - V_{z_\tau}\, W_{\tau-1}.$$

Because the noise matrices are independent and statistically invariant to multiplication by unitary matrices, we may write this as

$$X_\tau = V_{z_\tau}\, X_{\tau-1} + \sqrt{2}\, W'_\tau, \tag{6}$$

where W'_τ is a $M \times N$ matrix of additive independent $\mathcal{CN}(0, 1)$ noise. This is the *fundamental differential receiver equation.*

Remarkably, the matrix of fading coefficients H does not appear in the fundamental differential receiver equation (6). Formally this equation is simular to (4). The signal V_{z_τ} appears to be transmitted through a virtual channel with fading response $X_{\tau-1}$, which is *known* to the receiver, and corrupted by noise with twice the variance. This corresponds to the well known result that standard single-antenna differential modulation suffers from approximately a 3 dB performance loss in effective SNR when the channel is unknown versus when it is known.

Hence, it is not surprising that the maximum likelihood decoder is given by

$$\hat{z}_\tau = \arg \min_{\ell=0,\ldots,L-1} \|X_\tau - V_\ell X_{\tau-1}\| . \tag{7}$$

4 Probability of error

In the previous, we show section that the differential construction appears to turn the unknown-channel problem into a known-channel problem. But the construction still fundamentally involves an unknown channel. Because differential coding uses two consecutive blocks of M symbols, it is natural to look at blocks of $2M$ symbols within the framework of an unknown channel. The transmitted signals then look like

$$\begin{bmatrix} S_{\tau-1} \\ S_\tau \end{bmatrix} = \begin{bmatrix} S_{\tau-1} \\ V_{z_\tau} S_{\tau-1} \end{bmatrix} = \begin{bmatrix} I_m \\ V_\ell \end{bmatrix} S_{\tau-1}$$

Post-multiplying a signal by an $M \times M$ unitary matrix does not change the error performance of the signal [3]. The transmitted signal is therefore equivalent to the signal

$$\Phi_\ell = \frac{1}{\sqrt{2}} \begin{bmatrix} I_m \\ V_\ell \end{bmatrix} .$$

These symbols are normalized so that $\Phi_\ell^\dagger \Phi_\ell = I_M$. From this we see that the first block can be thought off as a training signal for the second block. The differential scheme post-multiplies Φ_ℓ with $S_{\tau-1}$ to let the previously transmitted block effectively act as the training signal for the current block.

It is shown in [3] that the probability of error of mistaking any two multiple-antenna signals Φ_ℓ and $\Phi_{\ell'}$ when the channel is unknown has Chernoff upper bound

$$P_e \leqslant \frac{1}{2} \prod_{m=1}^{M} \left[1 + \frac{\rho^2 (1 - d_m^2)}{1 + 2\rho} \right]^{-N} , \tag{8}$$

where $1 \geqslant d_1 \geqslant \ldots \geqslant d_M \geqslant 0$ are the singular values of the $M \times M$ correlation matrix $\Phi_\ell^\dagger \Phi_{\ell'}$ $(d_m = \sigma_m(\Phi_\ell^\dagger \Phi_{\ell'}))$. The pairwise probability of error P_e decreases as any d_m decreases. For large SNR, the error performance of a given constellation then is governed by the "diversity product"

$$\zeta = \min_{0 \leqslant \ell < \ell' \leqslant L-1} \left[\prod_{m=1}^{M} (1 - d_m^2) \right]^{1/M} . \tag{9}$$

Because $0 \leqslant d_m \leqslant 1$, we have $0 \leqslant \zeta \leqslant 1$. The product is raised to the power $1/M$ to get a quantity that is comparable for different M. The larger the ζ, the smaller the error probability, and any constellation with non-zero diversity product is said to have full diversity. The diversity product can also be expressed in terms of the V_ℓ matrices [6]

$$\zeta = \frac{1}{4} \min_{0 \leqslant \ell < \ell' \leqslant L-1} \left| \det \left(V_{\ell'} - V_\ell \right) \right|^{2/M} . \tag{10}$$

We also note that the the probability of mistaking any two signals V_ℓ and $V_{\ell'}$ in a non-differential setting *when the channel is known to the receiver* is bounded by

$$P_e \leqslant \frac{1}{2}\prod_{m=1}^{M}\left[1+\frac{\rho}{4}\sigma_m^2(V_{\ell'}-V_\ell)\right]^{-N}. \tag{11}$$

For high SNR, this bound is also governed by $|\det(V_{\ell'}-V_\ell)|$. Thus, matrices $V_0,\ldots,V_{L-1}$ that work well for differential modulation also work well when the channel is known, and vice-versa.

5 Group constellations

In this section, we assume that $\mathcal{V}$ forms a group. This entails some loss of generality because, in general, the product of any two matrices in $\mathcal{V}$ does not have to be another matrix in $\mathcal{V}$. The main advantage of a group constellation is that all transmitted matrices S_τ now belong to $\mathcal{V}$ as well. Differential modulation can now be written more succinctly by letting

$$S_\tau = V_{y_\tau} = V_{z_\tau}S_{\tau-1} \quad \text{where} \quad V_{y_\tau} = V_{z_\tau}V_{y_{\tau-1}}.$$

The transmitter does not need to compute the matrix multiplication explicitly, but simply can look up the group action in a stored table. Another advantage is simplified design. Good constellations are often found by searching over large candidate sets. Computing ζ for a general candidate constellation requires checking $(L-1)L/2$ determinants. However, when $\mathcal{V}$ is a group it suffices to check only $L-1$ determinants; in particular, one may check $|\det(I-V_\ell)|$.

To further simplify the construction we next assume that $\mathcal{V}$ is an Abelian (commutative) group. Imposing commutativity has some appealing consequences. Because $V_0,\ldots,V_{L-1}$ commute, they share a common set of eigenvectors. Consequently, this constellation of matrices can be diagonalized into a new constellation comprising diagonal matrices of eigenvalues using one fixed ℓ-independent similarity transform $V_\ell \to P^{-1}V_\ell P$. The similarity transform does not affect the error performance of the constellation. Thus, assuming $\mathcal{V}$ is Abelian is equivalent to assuming that all of its elements are diagonal matrices. This implies that at any given time only one antenna is active. We call these signals *diagonal.*

A commutative group can always be expressed as a product of cyclic groups. However, it is shown in [6] that only a single cyclic group by itself can lead to full diversity. We thus assume that $\mathcal{V}$ is cyclic, i.e., each element is a power of a generator Lth root of unity V_1: $V_\ell = V_1^\ell$. With such a group, matrix multiplication is then isomorphic to modulo-L addition on the indices, just as with standard DPSK. Hence, the transmitter does not even need a lookup table to compute the transmitted matrix. The matrix V_1 is diagonal and can be written as

$$V_1 = \begin{bmatrix} e^{i\frac{2\pi}{L}u_1} & 0 & \cdots \\ 0 & \ddots & 0 \\ 0 & \cdots & e^{i\frac{2\pi}{L}u_M} \end{bmatrix}, \quad u_m \in \{0,\ldots,L-1\}, \quad m = 1,\ldots,M.$$

These signals have a very simple interpretation. At any time, only one transmitter antenna is active and transmitting either a reference symbol (which in differential modulation is actually the previously sent symbol) or a phase-shifted symbol. Thus, within the τth block, antenna m transmits at time $t=\tau M+m$ a symbol that is differentially phase shifted by $(2\pi/L)u_m\ell$ relative to its previous transmission. The value of ℓ is determined by the data. It is important to

M	R	L	ζ	$[u_1\, u_2 \cdots u_M]$	P_e union bound ($\rho = 20$ dB)
1	1	2	1	[1] (standard DBPSK)	9.9e-3
2	1	4	0.5	[1 1]	1.7e-3
3	1	8	0.2636	[1 1 3]	4.6e-4
4	1	16	0.2973	[1 3 5 7]	6.7e-5
5	1	32	0.1677	[1 5 7 9 11]	3.0e-5
1	2	4	0.2929	[1] (standard DQPSK)	4.9e-2
2	2	16	0.1464	[1 7]	3.4e-2
3	2	64	0.07647	[1 11 27]	2.6e-2
4	2	256	0.04877	[1 25 97 107]	1.7e-2
5	2	1024	0.03998	[1 157 283 415 487]	9.1e-3

Table 1: Systematic antenna constellations for $M = 1, 2, 3, 4$ and 5 transmitter antennas and rate $R = 1, 2$ that maximize ζ in (10). The number of signals in the constellation is $L = 2^{RM}$. The P_e upper bound is a union bound on block error rate obtained by summing over $\ell \neq \ell'$ the Chernoff bounds (8) with $\rho = 20$ dB.

note that the phase shifts are potentially different for each antenna. When $M = 1$, the signals reduce to standard DPSK.

Constellation design is now reduced to finding u_m such that the diversity product of $\mathcal{V}$ is as large as possible. Because it is easy to find the determinant of the difference of two diagonal matrices, the design problem becomes

$$\{u_1, \ldots, u_M\} = \arg \max_{0 \leqslant u_1, \ldots, u_M \leqslant L-1} \min_{\ell = 1, \ldots, L-1} \left[\prod_{m=1}^{M} \sin^2(\pi u_m \ell / L) \right]^{1/M}.$$

We find good constellations through an exhaustive search. Several techniques can be used to reduce the exponentially large search space. In particular we can let $u_1 = 1$ and let the remaining u_m be relatively prime with respect to L. Furthermore, we may constrain $u_2, \ldots, u_M \in \{2, \ldots, L/2\}$.

Table 1 shows the results of our searches for constellations of $L = 2^{RM}$ that maximize ζ. Because L is a power of two, only odd u_m appear. For $M = 1$ transmitter antenna, the search naturally produces differential BPSK ($R = 1$) and differential QPSK ($R = 2$). Also included is an upper bound on the block error rate obtained by summing over $\ell \neq \ell'$ the Chernoff bounds (8) with $\rho = 20$ dB.

6 Results

We examine the performance of the constellations listed in Table 1 by sending the signals over a simulated continuously fading Rayleigh channel. The fading is assumed to be independent between antennas but correlated in time according to Jakes' model [7]. A typical physical scenario where such a model is appropriate is a base station antenna array communicating with a mobile with $N = 1$ receiver antenna. If we assume that the mobile is traveling at approximately 25 m/s (55 mph) and operating at 900 MHz, the Doppler shift is approximately $f_D = 75$ Hz. The Jakes correlation between two fading coefficients t time samples apart is $J_0(2\pi f_D T_s t)$, where T_s is the sampling period and J_0 is the zero-order Bessel function of the first kind. We assume that $T_s = 1/30,000$ so $T_s f_D = 0.0025$. The Jakes correlation function has its first zero at $t \approx 153$.

Figures 1 and 2 show the bit error performance of $M = 1, 2, 3, 4$, and 5 transmitter antennas and one receiver antenna for $R = 1$ and $R = 2$. We see that the differential unitary space-time signals are especially effective at high SNR. This is not inconsistent with claims in [3] that unitary space-time signals are best suited for high SNR. We also note that the block error union bounds presented in Table 1 give rough indications of the bit error performances shown in the figures. Because the fading is continuous, the effects of variations in the fading coefficients should be more apparent as the transmitted $M \times M$ signal matrices grow in dimension. Equivalently, the effects of fading should be more apparent as the number of transmitter antennas grows. This perhaps explains the limited gain in performance for $M = 5$ over $M = 4$ when $R = 1$, and the slight appearance of an error floor at very high SNR's.

An advantage of our diagonal signals is their simplicity. Because only one antenna transmits at any given time, one power amplifier can be switched among the antennas. But this amplifier must deliver M-times the power it would otherwise deliver if there were an array of M amplifiers simultaneously driving the other antennas. Consequently, this amplifier needs to have a larger linear operating range than an amplifier array would. Amplifiers with a large linear range are often expensive to design and build. It may therefore occasionally be desirable to have all M antennas transmitting simultaneously at lower power. In this case, we may modify the constellation by post-multiplying our signals by $M \times M$ unitary matrices such as discrete Fourier transform matrices. This has the effect of smearing the transmitted symbol on any active antenna across all of the antennas but does not affect the performance of the constellation at all.

References

[1] I. E. Telatar, "Capacity of multi-antenna Gaussian channels," Tech. Rep., Bell Laboratories, Lucent Technologies, 1995.

[2] G. J. Foschini, "Layered space-time architecture for wireless communication in a fading environment when using multi-element antennas," *Bell Labs. Tech. J.*, vol. 1, no. 2, pp. 41–59, 1996.

[3] B. M. Hochwald and T. L. Marzetta, "Unitary space-time modulation for multiple-antenna communication in Rayleigh flat-fading," Tech. Rep., Bell Laboratories, Lucent Technologies, 1998. Also to appear in *IEEE Trans. Info. Theory*.

[4] V. Tarokh, N. Seshadri, and A. R. Calderbank, "Space-time codes for high data rate wireless communication: Performance criterion and code construction," *IEEE Trans. Info. Theory*, vol. 44, pp. 744–765, 1998.

[5] V. Tarokh, H. Jafarkani, and A. R. Calderbank, "Space-time block codes from orthogonal designs," *IEEE Trans. Info. Theory*, vol. 45, pp. 1456–1467, July 1999.

[6] B. Hochwald and W. Sweldens, "Differential unitary space time modulation," Tech. Rep., Bell Laboratories, Lucent Technologies, March, 1999. Can be downloaded from `http://cm.bell-labs.com/who/wim/dust/`

[7] W. C. Jakes, *Microwave Mobile Communications*. Piscataway, NJ: IEEE Press, 1993.

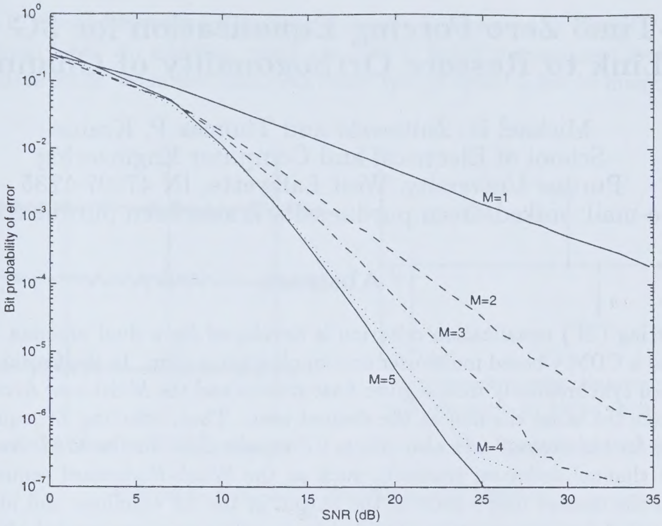


Figure 1: Performance of $M = 1$, 2, 3, 4, and 5 transmitter antennas and $N = 1$ receiver antenna as a function of SNR ρ. The data rate is $R = 1$. 1.

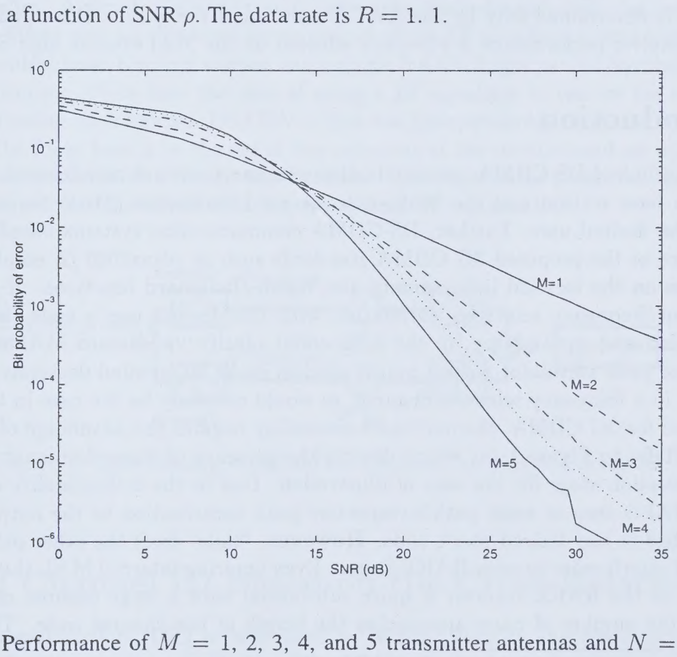


Figure 2: Performance of $M = 1$, 2, 3, 4, and 5 transmitter antennas and $N = 1$ receiver antenna as a function of SNR ρ. The data rate is $R = 2$. 1.

Space-Time Zero Forcing Equalization for 3G CDMA Forward Link to Restore Orthogonality of Channel Codes [1]

Michael D. Zoltowski and Thomas P. Krauss
School of Electrical and Computer Engineering
Purdue University, West Lafayette, IN 47907-1285
e-mail: mikedz@ecn.purdue.edu, krauss@ecn.purdue.edu

Abstract

A zero-forcing (ZF) equalization criterion is developed for a dual antenna receiver on the forward link in a CDMA based multi-user communication system. In the forward link, all users are broadcasted synchronously from a given base station and the Multi-user Access Interference (MAI) traverses the same channel as the desired user. Thus, effecting ZF equalization (with two antennas) for the desired user also effects ZF equalization for the MAI. Assuming the use of orthogonal channel codes on transmit, such as the Walsh-Hadamard sequences, one may correlate with the desired user's code at the output of the ZF equalizer and ideally have zero MAI in the symbol decision statistics obtained by sampling once per symbol after synchronization. Simulations will be presented demonstrating the advantages of the proposed two-channel ZF equalizer over a space-time RAKE receiver. The simulations show that the ZF equalizer performance is determined only by the AWGN noise (and the gain of the equalizer), whereas the RAKE receiver performance is adversely effected by the MAI even at high SNR.

1 Introduction

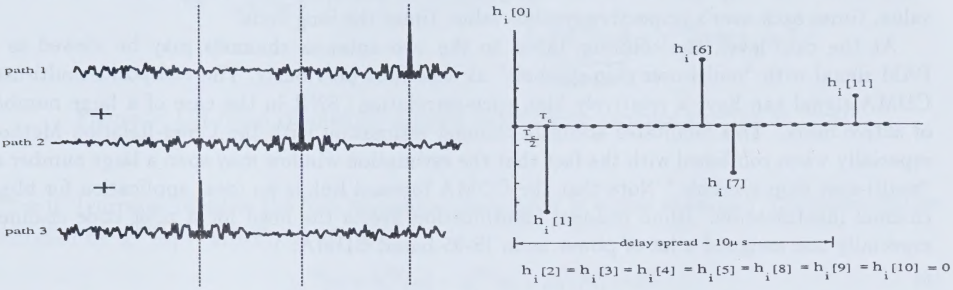
In the forward link of DS-CDMA communication systems, users are broadcasted synchronously from a given base station and the Multi-user Access Interference (MAI) traverses the same channel as the desired user. Further, DS-CDMA communication systems based on either the IS-95 standard or the proposed 3G CDMA standards such as cdma2000 [2] employ orthogonal channel codes on the forward link, namely, the Walsh-Hadamard functions. If the multipath channel is not frequency selective, correlation with the desired user's code (after achieving synchronization and multiplying by the long code) ideally yields zero MAI at each of the symbol-spaced peak correlator output points used to make bit/symbol decisions.

However, in a frequency selective channel, as would certainly be the case in the high-speed links proposed for 3G CDMA, the multipath ostensibly negates the advantage of using orthogonal codes. Refer to Figure 1 (a) which depicts the presence of three dominant multipaths in the post-correlation stage for the sake of illustration. Due to the orthogonality of the channel codes, the MAI is zero at **each** path's respective peak contribution to the output of the correlator matched to the desired user's code. **However**, "tails" from the other paths contribute MAI and self-interference to each RAKE finger. Even ignoring intercell MAI, the intracell MAI contribution to the RAKE receiver is quite substantial with a large number of active users, especially if the number of users approaches the length of the channel code. Thus, the BER

---

[1] This research was supported by the Air Force Office of Scientific Research under grant no. F49620-97-1-0275 and the National Science Foundation under grant no. MIPS-9708309.

curve for RAKE receiver flattens out as SNR increases as observed in Figure 6 which will be discussed in detail later. In addition, there are RAKE finger management issues, as well as issues of self-interference, when the multipath delay spread spans a few to many symbols.


path 1
+
path 2
+
path 3
$h_i[0]$
$h_i[6]$
$h_i[11]$
T_c
$\frac{T_c}{2}$
$h_i[7]$
$h_i[1]$
delay spread ~ 10μ s
$h_i[2] = h_i[3] = h_i[4] = h_i[5] = h_i[8] = h_i[9] = h_i[10] = 0$

(a) MAI contributions to RAKE receiver. (b) Sparse channel model.

Figure 1: Multipath Modelling and Issues.

Thus, we here propose the use of Zero-Forcing (ZF) equalization in conjunction with a dual receiver antenna at the mobile hand-set, as depicted in Figure 2, to restore the orthogonality of the channel codes. A simple correlator matched to the desired user's code (or, equivalently, a code-matched filter) may be employed at the output of the ZF equalizer to make bit/symbol decisions. In the ideal case, the final output has a single RAKE finger per bit/symbol period with no MAI contribution. Note that the idea of using a ZF equalizer to restore the orthogonality of the channel codes on a forward DS-CDMA link was independently proposed by Ghauri and Slock in [1]. The focus here is on the use of two antennas at the mobile hand-set either spatially separated or co-located with different polarizations. The algorithms presented are tailored for a dual antenna receiver thereby facilitating space-time ZF equalization.

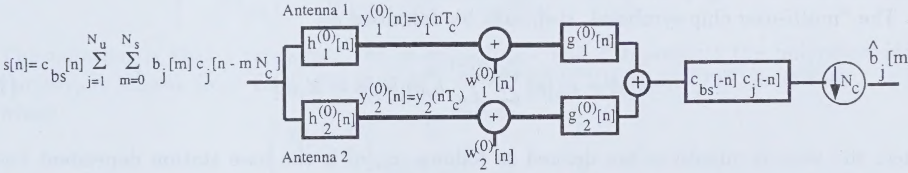

Antenna 1
$y_1^{(0)}[n]=y_1(nT_c)$
$s[n]= c_{bs}[n] \sum_{j=1}^{N_u} \sum_{m=0}^{N_s} b_j[m]\, c_j[n - m N_c]$
$h_1^{(0)}[n]$
+
$w_1^{(0)}[n]$
$g_1^{(0)}[n]$
+
$c_{bs}[-n]\, c_j[-n]$
N_c
$\hat{b}_j[m]$
$y_2^{(0)}[n]=y_2(nT_c)$
$h_2^{(0)}[n]$
+
$g_2^{(0)}[n]$
Antenna 2
$w_2^{(0)}[n]$

Figure 2: Zero-Forcing Equalization Followed by Simple Correlator.

2 Blind Channel ID Based on the Cross-Relation

The SIMO Cross-Relation (CR) is here used to blindly identify the respective impulse response for each antenna channel [3, 4]. The CR Method is based on the following *deterministic* relationship that holds when the same signal is input to two different FIR filters:

$y_i[n] * h_j[n] - y_j[n] * h_i[n] = 0$, where $h_i[n]$ is the impulse response of the *i-th* channel and $y_i[n]$ is the output of the *i-th* channel. The CR Method can be shown to be an approximate ML estimator of the channel impulse responses under the following assumptions: (i) the receiver noise is Gaussian and (ii) the "multi-user chip symbols" are deterministic but unknown. Each "multi-user chip symbol" is the superposition of each active user's respective channel code value, times each user's respective symbol value, times the long code.

At the chip level, the common input to the two antenna channels may be viewed as a PAM signal with "multi-user chip symbols" as described previously. The composite multi-user CDMA signal can have a relatively high (pre-correlation) SNR in the case of a large number of active users. This facilitates accurate channel estimation with the Cross-Relation Method especially when combined with the fact that the estimation window may span a large number of "multi-user chip symbols." Note that the CDMA forward link is an ideal application for blind channel identification. Blind channel identification averts the need for a pilot code channel, especially one assigned a lot of power as in IS-95 based CDMA.

2.1 Channel Modelling and Blind Channel ID

For the sake of notational simplicity (and for comparisons with RAKE receiver performance), we here initially assume a high enough chip rate so that the multipath delays are integer multiples of the chip time T_c. In addition, we initially assume the received signal at each antenna is sampled at twice the chip rate and convolved with the sampled chip waveform having a square-root raised cosine spectrum. The impulse response for the *i-th* antenna channel including chip pulse shaping is then given by

$$h_i(t) = \sum_{k=1}^{L} h_i[k] p_{rc}(t - kT_c) \quad i = 1, 2 \tag{1}$$

where $p_{rc}(t)$ has a raised-cosine spectrum. Note that $h_i[k]$ may be zero for many path delays kT_c in a high-speed case where the channel is sparse as shown in Figure 1 (b). LT_c is the multipath delay spread; the delay spread in a typical urban cellular environment has been experimentally determined to be roughly seven to ten microseconds.

The "multi-user chip symbols", $s[n]$, may be described as

$$s[n] = c_{bs}[n] \sum_{j=1}^{N_u} \sum_{m=0}^{N_s-1} b_j[m] c_j[n - N_c m] \tag{2}$$

where the various quantities are defined as follows: $c_{bs}[n]$ is the base station dependent long code; $b_j[m]$ is the *j-th* user's bit/symbol sequence; $c_j[n]$, $n = 0, 1, ..., N_c - 1$, is the *j-th* user's channel (short) code; N_c is the length of each channel code; N_u is the total number of active users; N_s is the number of bit/symbols transmitted during a given time window. The signal received at the *i-th* antenna (after convolving with a filter impulse response having a square-root raised cosine spectrum) may be described as

$$y_i(t) = \sum_n s[n] h_i(t - nT_c) \quad i = 1, 2 \tag{3}$$

where $h_i(t)$ is defined in Eqn. (1). Consider chip rate sampling of $y_i(t)$ in Eqn. (3) starting at

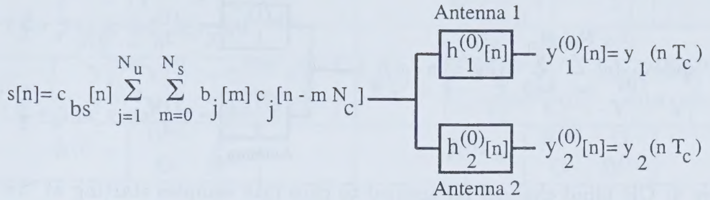


Figure 3: CR blind channel ID applied to chip rate samples starting at "t=0".

"$t=0$" (corresponding to the first polyphase component at either antenna.) The cross-relation satisfied by $y_i^{(0)}[n]=y_i(nT_c)$, $i=1,2$, is

$$y_2^{(0)}[n]*h_1^{(0)}[n]-y_1^{(0)}[n]*h_2^{(0)}[n]=0. \tag{4}$$

where the respective impulse responses $h_1^{(0)}[n]$ and $h_2^{(0)}[n]$ are given by

$$h_i^{(0)}[n]=\sum_{k=0}^{L} h_i[k]\,p_{rc}^{(0)}[n-k] \tag{5}$$

with $p_{rc}^{(0)}[n]=p_{rc}[2n]=\delta[n]$, assuming the autocorrelation of the chip waveform to have a raised cosine spectrum. Here $p_{rc}[n]$ is $p_{rc}(t)$ sampled at twice chip rate: $p_{rc}[n]=p_{rc}\left(n\frac{T_c}{2}\right)$.

The Cross-Relation in Eqn. (4) for each of N_s chip-rate samples may be compiled in matrix form as $\left[\mathbf{Y}_2^{(0)}\vdots-\mathbf{Y}_1^{(0)}\right]\begin{bmatrix}\mathbf{h}_1^{(0)}\\ \mathbf{h}_2^{(0)}\end{bmatrix}=\mathbf{0}$, where

$$\mathbf{Y}_i^{(\ell)}=\begin{bmatrix} y_i^{(\ell)}[L] & y_i^{(\ell)}[L-1] & \cdots & y_i^{(\ell)}[0] \\ y_i^{(\ell)}[L+1] & y_i^{(\ell)}[L] & \cdots & y_i^{(\ell)}[1] \\ \vdots & \vdots & & \vdots \\ y_i^{(\ell)}[N_s] & y_i^{(\ell)}[N_s-1] & \cdots & y_i^{(\ell)}[N_s-L] \end{bmatrix}, \quad i=1,2, \quad \ell=0,1 \tag{6}$$

The notation is that i represents the antenna index and ℓ represents the polyphase index. Further, it follows from Eqn. (5) that $\mathbf{h}_i^{(0)}=[h_i^{(0)}[0],h_i^{(0)}[1],...,h_i^{(0)}[L]]^T=\mathbf{B}^{(0)}\mathbf{h}_i$, $i=1,2$, where

$$\mathbf{B}^{(\ell)}=\begin{bmatrix} p_{rc}^{(\ell)}[-L'] & 0 & \dots & 0 \\ \vdots & p_{rc}^{(\ell)}[-L'] & 0 & 0 \\ p_{rc}^{(\ell)}[0] & \vdots & \ddots & \vdots \\ \vdots & p_{rc}^{(\ell)}[0] & \ddots & p_{rc}^{(\ell)}[-L'] \\ p_{rc}^{(\ell)}[L'] & \vdots & \ddots & \vdots \\ 0 & p_{rc}^{(\ell)}[L'] & \ddots & p_{rc}^{(\ell)}[0] \\ 0 & 0 & \ddots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & p_{rc}^{(\ell)}[L'] \end{bmatrix} \quad \ell=0,1 \tag{7}$$

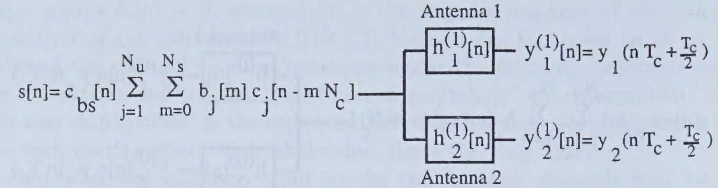

Antenna 1
$s[n]=c_{bs}[n]\sum_{j=1}^{N_u}\sum_{m=0}^{N_s} b_j[m]\, c_j[n-mN_c]$
$h_1^{(1)}[n]$
$y_1^{(1)}[n]=y_1(nT_c+\frac{T_c}{2})$
$h_2^{(1)}[n]$
$y_2^{(1)}[n]=y_2(nT_c+\frac{T_c}{2})$
Antenna 2

Figure 4: CR blind channel ID applied to chip rate samples starting at "t=$T_c/2$".

The cross-relation satisfied by $y_i^{(1)}[n] = y_i(nT_c + T_c/2)$, $i = 1, 2$, is

$$y_2^{(1)}[n] * h_1^{(1)}[n] - y_1^{(1)}[n] * h_2^{(1)}[n] = 0 \tag{8}$$

where the respective impulse responses are

$$h_i^{(1)}[n] = \sum_{k=0}^{L} h_i[k] p_{rc}^{(1)}[n-k] \tag{9}$$

with $p_{rc}^{(1)}[n] = p_{rc}[2n+1] \neq \delta[n]$, where, again, $p_{rc}[n]$ is $p_{rc}(t)$ sampled at twice chip rate. In matrix form, we have $\left[\mathbf{Y}_2^{(1)} \vdots -\mathbf{Y}_1^{(1)}\right] \begin{bmatrix} \mathbf{h}_1^{(1)} \\ \mathbf{h}_2^{(1)} \end{bmatrix} = \mathbf{0}$ where $\mathbf{h}_i^{(1)} = \mathbf{B}^{(1)}\mathbf{h}_i$, $i = 1, 2$.

The composite channel vector $\mathbf{h} = \begin{bmatrix} \mathbf{h}_1 \\ \mathbf{h}_2 \end{bmatrix}$, where $\mathbf{h}_i = [h_i[0], h_i[1], \ldots, h_i[L]]^T \quad i = 1, 2$, may be estimated as the "smallest" right singular vector of either $\left[\mathbf{Y}_2^{(0)}\mathbf{B}^{(0)} \vdots -\mathbf{Y}_1^{(0)}\mathbf{B}^{(0)}\right]$ or $\left[\mathbf{Y}_2^{(1)}\mathbf{B}^{(1)} \vdots -\mathbf{Y}_1^{(1)}\mathbf{B}^{(1)}\right]$. The two estimates may be combined in an incoherent fashion, along with subtracting out an estimate of the noise power $\hat{\sigma}_n^2$ from the diagonal terms, by estimating $\mathbf{h}$ as that vector which minimizes the quadratic form

$$\mathbf{h}^H \left\{ \tilde{\mathbf{B}}^{(0)H} [\mathbf{R}_Z^{(0)} - \hat{\sigma}_n^2 \mathbf{I}] \tilde{\mathbf{B}}^{(0)} + \tilde{\mathbf{B}}^{(1)H} [\mathbf{R}_Z^{(1)} - \hat{\sigma}_n^2 \mathbf{I}] \tilde{\mathbf{B}}^{(1)} \right\} \mathbf{h} \tag{10}$$

subject to some constraint. The various matrices in Eqn. (10) are defined in Table 1 and follow from the previous developments. Employing a unity norm constraint, $\mathbf{h}$ is estimated as the "smallest" eigenvector of $\tilde{\mathbf{B}}^{(0)H}[\mathbf{R}_Z^{(0)} - \hat{\sigma}_n^2\mathbf{I}]\tilde{\mathbf{B}}^{(0)} + \tilde{\mathbf{B}}^{(1)H}[\mathbf{R}_Z^{(1)} - \hat{\sigma}_n^2\mathbf{I}]\tilde{\mathbf{B}}^{(1)}$.

2.2 ZF Equalizer Design

The synthesis of the equalizing filters $g_1^{(0)}[n]$ and $g_2^{(0)}[n]$, $n = 0, 1, ..., L$, depicted in Figure 2 is based on minimizing the output noise power subject to the ZF constraint. Mathematically,

$$\underset{g_1^{(0)}[n],\, g_2^{(0)}[n]}{\text{Minimize}} \quad \sum_{n=0}^{L} \{ |g_1^{(0)}[n]|^2 + |g_2^{(0)}[n]|^2 \}$$

$$\text{subject to:} \quad h_1^{(0)}[n] * g_1^{(0)}[n] + h_2^{(0)}[n] * g_2^{(0)}[n] = \delta[n-D] \tag{11}$$

$\mathbf{Z}^{(0)} = \boxed{\mathbf{Y}_2^{(0)} \vdots -\mathbf{Y}_1^{(0)}}$	$\mathbf{Z}^{(1)} = \boxed{\mathbf{Y}_2^{(1)} \vdots -\mathbf{Y}_1^{(1)}}$
$\mathbf{R}_Z^{(0)} = \mathbf{Z}^{(0)H}\mathbf{Z}^{(0)}$	$\mathbf{R}_Z^{(1)} = \mathbf{Z}^{(1)H}\mathbf{Z}^{(1)}$
$\mathbf{R}_Z^{(0)} = \begin{bmatrix} \mathbf{Y}_2^{(0)H}\mathbf{Y}_2^{(0)} & -\mathbf{Y}_2^{(0)H}\mathbf{Y}_1^{(0)} \\ -\mathbf{Y}_1^{(0)H}\mathbf{Y}_2^{(0)} & \mathbf{Y}_1^{(0)H}\mathbf{Y}_1^{(0)} \end{bmatrix}$	$\mathbf{R}_Z^{(1)} = \begin{bmatrix} \mathbf{Y}_2^{(1)H}\mathbf{Y}_2^{(1)} & -\mathbf{Y}_2^{(1)H}\mathbf{Y}_1^{(1)} \\ -\mathbf{Y}_1^{(1)H}\mathbf{Y}_2^{(1)} & \mathbf{Y}_1^{(1)H}\mathbf{Y}_1^{(1)} \end{bmatrix}$
$\tilde{\mathbf{B}}^{(0)} = \boxed{\begin{matrix} \mathbf{B}^{(0)} & \mathbf{O} \\ \mathbf{O} & \mathbf{B}^{(0)} \end{matrix}}$	$\tilde{\mathbf{B}}^{(1)} = \boxed{\begin{matrix} \mathbf{B}^{(1)} & \mathbf{O} \\ \mathbf{O} & \mathbf{B}^{(1)} \end{matrix}}$

Table 1: Prescription for Construction of Matrices Employed in Eqn. (10).

where D is some integer chip delay. That is, ZF equalization may be effected across the two antennas by combining the *chip-spaced* samples $y_i^{(0)}[n]$, $i = 1, 2$. In matrix form, the ZF constraint may be expressed as $\mathcal{H} = \left[\mathbf{H}_1^{(0)} : \mathbf{H}_2^{(0)}\right] \begin{bmatrix} \mathrm{g}_1^{(0)} \\ \mathrm{g}_2^{(0)} \end{bmatrix} = \boldsymbol{\delta}$ where $\boldsymbol{\delta}$ is all zeroes except for unity in the $D-th$ position and $\mathbf{H}_i$ is the convolution matrix

$$\mathbf{H}_i = \begin{bmatrix} h_i^{(0)}[0] & 0 & \cdots & 0 \\ h_i^{(0)}[1] & h_i^{(0)}[0] & 0 & 0 \\ \vdots & \ddots & \ddots & \ddots \\ h_i^{(0)}[L] & h_i^{(0)}[L\text{-}1] & \ddots & h_i^{(0)}[0] \\ 0 & h_i^{(0)}[L] & \ddots & h_i^{(0)}[1] \\ \vdots & & \ddots & \ddots \\ 0 & 0 & 0 & h_i^{(0)}[L] \end{bmatrix} \tag{12}$$

The solution is simply $\mathrm{g} = \mathcal{H}^{\dagger}\boldsymbol{\delta}$, where $\mathcal{H}^{\dagger}$ is the pseudo-inverse of $\mathcal{H}$.

3 Simulations

A wideband CDMA forward link was simulated similar to one of the options in the US cdma2000 proposal [2]. Simulations were performed for 16, 32 and 64 intra-cell users. The chip rate was

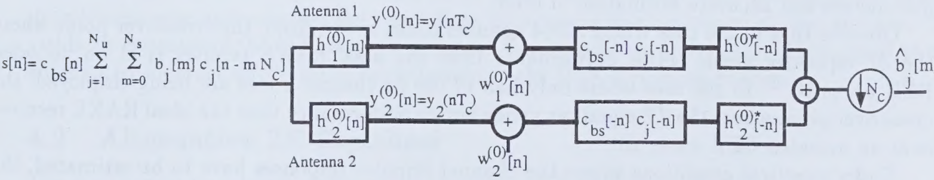


Figure 5: RAKE Receiver with Maximal Ratio Combining.

3.6864 MHz, 3 times that of IS-95, which is one of the rates proposed for cdma2000. The spreading factor was $Nc = 64$ chips per bit. The data symbols were BPSK which, for each user, was spread with a length 64 Walsh-Hadamard function. The sum signal was scrambled with a multiplicative QPSK spreading sequence ("scrambling code") of length 32768 similar to the IS-95 standard (in IS-95, the offset into this code identifies the base station). The signals for all the users were added synchronously.

For the simulations presented here, the channel was assumed known. BER results were obtained by averaging over many different channels and among the different users for varying SNRs. For each trial run, a channel was generated consisting of 4 arrival paths with propagation delays randomly spaced in the interval $[0, \tau_{max} = 10\mu sec \approx 36\text{chips}]$. The arrival times were the same for each of the two receive antennas, and were chosen as the first 4 elements of a random permutation of the set $\{0, 1, ..., 36\}$ (in chips). The real and imaginary part of the complex gain for each path were independent normally distributed random variables with zero mean and unit variances; the gains were independent across antennas as well as across paths.

As depicted in Figure 5, the RAKE receiver employed a maximal ratio combiner to coherently combine the 4 different paths from both antennas. The ZF equalizing filter at either antenna had a length of 37 chips (equal to the maximum delay spread). The delay of the signal through both channel and ZF equalizer was set to 39 chips. The results of the simulations are plotted in Figure 6. Note that each of the SNR values forming the abscissa in Figure 6. is the **post**-correlation SNR for the desired user which includes a processing gain of $10\log(64) \approx 18$ dB. The effects of inter-cell interference were not simulated.

The scenario was set up to provide the best possible conditions for a RAKE receiver employing maximal ratio combining to perform well. The complex gain of each multipath at either antenna was perfectly known and the peaks of the code-matched correlator outputs were lined up with chip-sampled points. In addition, the delay spread was only about a half-symbol in duration thereby making the effects of self-interference negligible. Notwithstanding, note that the performance of the RAKE receiver is heavily dependent on the number of users. For each of the three levels of user numbers simulated, the BER curve of the RAKE receiver is observed to flatten out above a certain SNR. In contrast, the performance of the ZF equalizer followed by a simple correlator is observed to be invariant with respect to the number of users.

The RAKE receiver benefits from diversity gains at lower SNR, while the ZF equalizer benefits from the orthogonality of codes at moderate to high SNR. Because each of the four paths has the same strength at each of the two antennas, and all paths have at least a chip spacing between them, the RAKE receiver benefits from roughly a $10\log(8) = 9$ dB diversity gain. Note that in practice the achievable diversity gain depends heavily on the actual multipath parameters and accurate estimation of them.

Observe that in the case where all 64 channel codes are utilized, the cross-over point where the ZF equalizer yields better performance than the ideal RAKE receiver is at an uncoded BER *above* 10^{-2}. In the case where only half of the 64 channel codes are being employed, the cross-over point where the ZF equalizer yields better performance than the ideal RAKE receiver is at an uncoded BER *above* 10^{-3}.

Under practical conditions where the channel impulse responses have to be estimated, the performance of the ZF equalizer will gracefully decline. The RAKE receiver performance is much more sensitive to proper phasing of RAKE fingers and sampling issues. In addition, there are RAKE finger management issues, as well as issues of self-interference when the delay spread spans a few to many symbols.

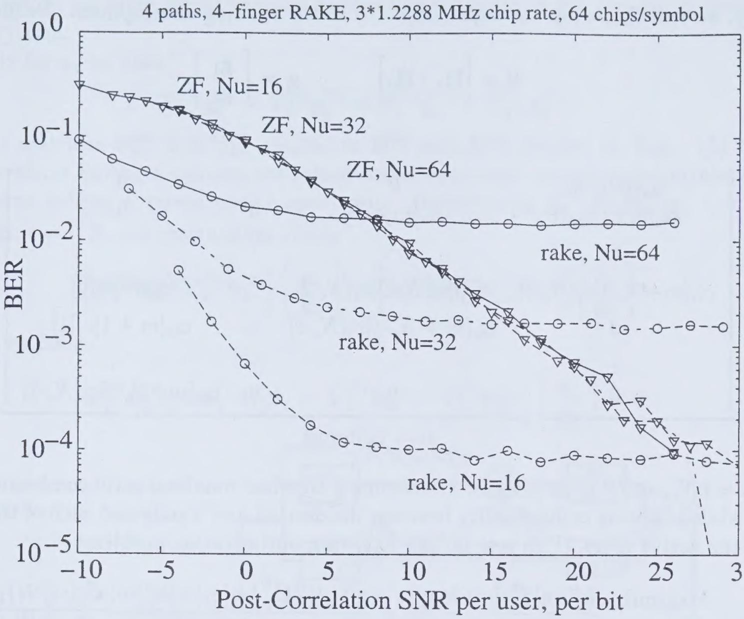


Figure 6: RAKE receiver versus ZF Equalizer for Varying Numbers of Users.

4 Future Work

4.1 Hybrid RAKE Receiver

The RAKE receiver benefits from diversity gains at lower SNR, while the ZF equalizer benefits from the orthogonality of codes at moderate to high SNR. This suggests a Hybrid ZF equalizer/RAKE Receiver which might operate as follows. First, employ the ZF Equalizer to make initial decisions on the desired user's symbols. Second, reconstruct the desired user's transmitted signal including it's code and multiplication by the proper portion of the long code. Third, for each antenna, convolve the reconstructed desired user's signal with the respective channel impulse response and subtract the result from the original received signal at that antenna. This yields an estimate of the MAI at each antenna. Employ this estimate of the MAI signal at each antenna to subtract off the MAI contribution to the RAKE receiver output.

4.2 Alternative ZF Equalizer

The goal here is to achieve performance closer to the RAKE receiver when a smaller number of channel codes are actively being used. The idea is to only require orthogonality of codes to be restored for active users. Note, $h_i^{(0)}[n] = h_i[n]$ when the autocorrelation of the chip waveform has a raised cosine spectrum since $p_{rc}^{(0)}[n] = p_{rc}[2n] = \delta[n]$. For the purpose of notational simplicity, we will thus drop the superscript (0) in the ensuing development. Define

$$\mathcal{H} = \left[\mathbf{H}_1 \vdots \mathbf{H}_2\right] \qquad \mathbf{g} = \left[\begin{array}{c} \mathbf{g}_1 \\ \mathbf{g}_2 \end{array}\right] \tag{13}$$

where

$$\mathcal{C}_j[m] = \underbrace{\left[\begin{array}{cccc} c_{bs}[m]c_j[0] & 0 & \cdots & 0 \\ c_{bs}[m+1]c_j[1] & c_{bs}[m]c_j[0] & 0 & 0 \\ \vdots & \ddots & \ddots & \ddots \\ c_{bs}[m+N_c\text{-}1]c_j[N_c\text{-}1] & c_{bs}[m+N_c\text{-}2]c_j[N_c\text{-}2] & \ddots & c_{bs}[m]c_j[0] \\ 0 & c_{bs}[m+N_c\text{-}1]c_j[N_c\text{-}1] & \ddots & c_{bs}[m+1]c_j[1] \\ \vdots & \ddots & \ddots & \ddots \\ 0 & 0 & 0 & c_{bs}[m+N_c\text{-}1]c_j[N_c\text{-}1] \end{array}\right]}_{2L\text{-}1 \text{ columns}} \quad j = 1, ..., N_u \tag{14}$$

where $m = kN_c$, and k is an integer. Determine g to effect maximal ratio combining as best as possible while restoring orthogonality between the desired user's code and each of the respective codes of the active users. This sets up the following optimization problem:

$$\underset{\mathbf{g}}{\text{Maximize}} \; \frac{|\tilde{\mathbf{c}}_1^H[m]\; \mathcal{C}_1[m]\; \mathcal{H}\; \mathbf{g}|^2}{\mathbf{g}^H\mathbf{g}} = \frac{\mathbf{g}^H\{\mathcal{H}^H\mathcal{C}_1^H[m]\tilde{\mathbf{c}}_1[m]\tilde{\mathbf{c}}_1^H[m]\;\mathcal{C}_1[m]\;\mathcal{H}\}\mathbf{g}}{\mathbf{g}^H\mathbf{g}} \tag{15}$$

$$\text{subject to:} \;\; \tilde{\mathbf{c}}_1^H[m]\; \mathcal{C}_j[m]\; \mathcal{H}\; \mathbf{g} = 0 \quad j = 2, ..., N_u \tag{16}$$

where

$$\tilde{\mathbf{c}}_1^H[m] = [\mathbf{0}^T \; \{\mathbf{c}_{bs}[m] \odot \mathbf{c}_1\}^H \; \mathbf{0}^T] \tag{17}$$

The solution is

$$\mathbf{g}_{opt} = \mathbf{P}_D^{\perp}\{\mathcal{H}^H\mathcal{C}_1^H[m]\tilde{\mathbf{c}}_1[m]\} \quad \text{where:} \quad \mathbf{P}_D^{\perp} = \mathbf{D}(\mathbf{D}^H\mathbf{D})^{-1}\mathbf{D}^H \tag{18}$$

$$\mathbf{D} = \left[\mathcal{H}^H\mathcal{C}_2^H[m]\tilde{\mathbf{c}}_1[m] \vdots \mathcal{H}^H\mathcal{C}_3^H[m]\tilde{\mathbf{c}}_1[m] \vdots \ldots \vdots \mathcal{H}^H\mathcal{C}_{N_u}^H[m]\tilde{\mathbf{c}}_1[m]\right] \tag{19}$$

4.3 Direct Two-Channel Zero-Forcing Equalization

ZF equalization may be effected across the two antennas with chip-rate samples obtained starting at "t=0". ZF equalization may also be effected across the two antennas with chip-rate samples obtained starting at "t=$\frac{T_c}{2}$". The output should be the same in either case equal to the originally transmitted "multi-user chip-level symbols". Mathematically, we have

$$y_1^{(0)}[n] * g_1^{(0)}[n] + y_2^{(0)}[n] * g_2^{(0)}[n] = y_1^{(1)}[n] * g_1^{(1)}[n] + y_2^{(1)}[n] * g_2^{(1)}[n] \tag{20}$$

An important observation is that the coefficients characterizing the equalizing filters $g_1^{(0)}[n]$ and $g_2^{(0)}[n]$ for the "t=0" starting point are the *same* as those characterizing the equalizing filters $g_1^{(1)}[n]$ and $g_2^{(1)}[n]$ for the "t=$\frac{T_c}{2}$" starting point - only the basis functions change! The basis functions depend on the transmitted chip waveform and are therefore pre-determined. These principles lead to an error signal thereby facilitating tracking of the equalizing filters via an LMS-type system.

In matrix form, we have

$$\mathbf{Y}_1^{(0)}\mathbf{g}_1^{(0)} + \mathbf{Y}_2^{(0)}\mathbf{g}_2^{(0)} = \mathbf{Y}_1^{(1)}\mathbf{g}_1^{(1)} + \mathbf{Y}_2^{(1)}\mathbf{g}_2^{(1)} \tag{21}$$

Define basis matrices $\mathbf{B}_{sr}^{(0)}$ and $\mathbf{B}_{sr}^{(1)}$ similar to $\mathbf{B}^{(0)}$ and $\mathbf{B}^{(1)}$ defined by Eqn. (7) but using the chip waveform having a square-root raised-cosine spectrum, not the autocorrelation of the chip waveform having a raised cosine spectrum. Substituting $\mathbf{g}_i^{(0)} = \mathbf{B}_{sr}^{(0)}\mathbf{g}_i$, $i = 1, 2$ and $\mathbf{g}_i^{(1)} = \mathbf{B}_{sr}^{(1)}\mathbf{g}_i$, $i = 1, 2$, and rearranging yields

$$\left[\mathbf{Y}_1^{(0)}\mathbf{B}_{sr}^{(0)} \vdots \mathbf{Y}_2^{(0)}\mathbf{B}_{sr}^{(0)}\right]\left[\begin{array}{c}\mathbf{g}_1\\ \mathbf{g}_2\end{array}\right] = \left[\mathbf{Y}_1^{(1)}\mathbf{B}_{sr}^{(1)} \vdots \mathbf{Y}_2^{(1)}\mathbf{B}_{sr}^{(1)}\right]\left[\begin{array}{c}\mathbf{g}_1\\ \mathbf{g}_2\end{array}\right] \tag{22}$$

Ultimately, we have

$$\left[\mathbf{Y}_1^{(0)}\mathbf{B}_{sr}^{(0)} - \mathbf{Y}_1^{(1)}\mathbf{B}_{sr}^{(1)} \ \vdots \ \mathbf{Y}_2^{(0)}\mathbf{B}_{sr}^{(0)} - \mathbf{Y}_2^{(1)}\mathbf{B}_{sr}^{(1)}\right]\left[\begin{array}{c}\mathbf{g}_1\\ \mathbf{g}_2\end{array}\right] = \mathbf{0} \tag{23}$$

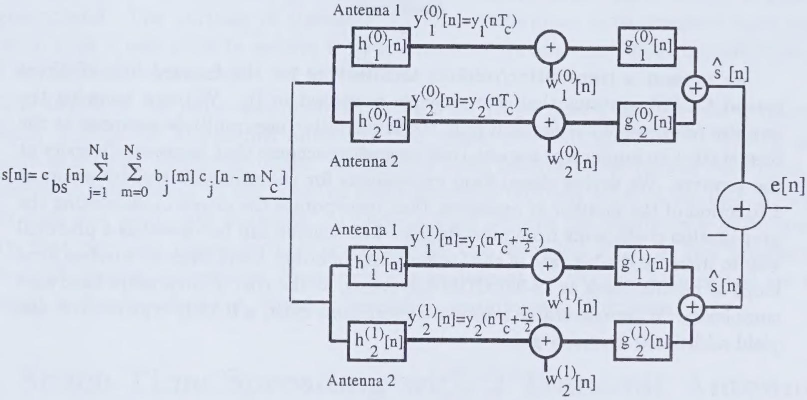

Antenna 1
$y_1^{(0)}[n]=y_1(nT_c)$
$h_1^{(0)}[n]$
+
$g_1^{(0)}[n]$
$\hat{s}[n]$
$w_1^{(0)}[n]$
+
$y_2^{(0)}[n]=y_2(nT_c)$
$h_2^{(0)}[n]$
+
$g_2^{(0)}[n]$
$w_2^{(0)}[n]$
Antenna 2
$s[n]= c_{bs}[n] \sum_{j=1}^{N_u} \sum_{m=0}^{N_s} b_j[m]\, c_j[n - m N_c]$
e[n]
+
—
Antenna 1
$y_1^{(1)}[n]=y_1(nT_c+\frac{T_c}{2})$
$h_1^{(1)}[n]$
+
$g_1^{(1)}[n]$
$\hat{s}[n]$
$w_1^{(1)}[n]$
+
$y_2^{(1)}[n]=y_2(nT_c+\frac{T_c}{2})$
$h_2^{(1)}[n]$
+
$g_2^{(1)}[n]$
Antenna 2
$w_2^{(1)}[n]$

Figure 7: Direct Blind ZF Equalizer.

References

[1] I. Ghauri and DTM. Slock, "Linear receivers for the DS-CDMA downlink exploiting orthogonality of spreading sequences," *Conf. Record of 32nd Asilomar Conf. on Signals, Systems and Computers*, Pacific Grove, CA, pp.650-654, Nov. 1998.

[2] The cdma2000 ITU-R RTT Candidate Submission, Telecommunications Industry Association, 27 July 1998.

[3] G. Xu, H. Liu, L. Tong & T. Kailath, "A Least-Squares Approach to Blind Channel Identification," *IEEE Trans. on Signal Processing,* pp. 2982-2993, Dec. 1995.

[4] M. D. Zoltowski, D. Tseng, and T. A. Thomas, "On the Use of Basis Functions in Blind Equalization Based on Deterministic Least Squares," *Proc. Asilomar Conference on Signals, Systems, and Computers*, Pacific Grove, CA, pp. 816-822, Nov. 1997.

A Novel Space-Time Spreading Scheme for Wireless CDMA Systems

B. M. Hochwald T. L. Marzetta C. B. Papadias

Bell Labs Research, Lucent Technologies
New Jersey, U.S.A.
{hochwald,marzetta,papadias}@bell-labs.com

Abstract

We present a transmitter/receiver architecture for the forward link of direct spread CDMA systems that was recently proposed in [1]. Without knowing the impulse response of the forward link, the transmitter uses multiple antennas at the base station to implement a space-time spreading scheme that improves diversity at the receiver. We derive closed-form expressions for the probability of bit error, as a function of the number of antennas, that incorporate the errors in estimating the propagation coefficients from pilot signals. The scheme can be viewed as a practical way to increase the bit rate of the forward link of either fixed (such as wireless local loop) or mobile (such as IS-95) CDMA systems, at the cost of little extra hardware complexity. Where favorable multipath conditions exist, a RAKE-type receiver can yield additional diversity gains.

1 Introduction

There has been recent interest in boosting the capacity and data rate of existing CDMA systems by using multiple base station antennas. It is generally accepted that multiple antennas add diversity, or independent paths, to a communication link, but multiple antennas currently appear to be more practical to build at a base station than on a hand-held mobile. On the uplink (from mobile to base), multiple antennas at the base station are successfully used with maximum ratio combining (MRC) to improve uplink data rates and to lower error probabilities. On the downlink, however, multiple antennas present some challenges. In this paper we will present a downlink scheme that provides important diversity gains without requiring significant additional resources at the base station.

In order to simplify the notation and focus on the multiple-antenna aspect, we pretend for the moment that we are using a CDMA system with only two users experiencing no delay spread. We would like to transmit respective real data symbols b_1 and b_2 to users 1 and 2, where b_k is multiplied by its own spreading code $\underline{c}_k$. The spreading codes are assumed to be orthogonal $\underline{c}_1^\dagger \underline{c}_2 = 0$, and to have unit norm $\|\underline{c}_1\| = \|\underline{c}_2\| = 1$ (throughout the paper, T denotes transpose and $\dagger$ denotes Hermitian transpose). The length of $\underline{c}_k$ is defined as the spreading gain, and it is understood that the elements of $\underline{c}_k$ are transmitted at the chip rate. The exact spreading codes, gains, and pulse shapes are, for our purposes, unimportant. Employing one transmitter antenna at the base station yields the received baseband signal at the kth user,

$$\underline{r}_k = h(b_1\underline{c}_1 + b_2\underline{c}_2) + \underline{n}, \qquad k = 1, 2, \tag{1}$$

where h is a complex-Gaussian fading coefficient having Rayleigh distributed amplitude and uniformly distributed phase, and $\underline{n}$ is a $P \times 1$ vector of additive zero-mean complex-Gaussian receiver noise samples. Although the fading coefficient h and additive noise $\underline{n}$ depend on k, to minimize clutter we omit its notational dependence. After being despread, the received signal becomes

$$d_k = hb_k + \nu, \qquad k = 1, 2, \tag{2}$$

where $\nu = \underline{c}_k^\dagger \underline{n}$. We assume that the mobile receiver knows h, but good decoding of b_k usually depends on the channel gain $|h|$ being reasonably large. This cannot, in general, be guaranteed. The purpose of transmit diversity techniques is to transmit each user's signal in such a way so as to deliver it to the receiver through multiple channel branches that are not fully correlated.

The sequel of the paper is organized as follows. In Section 2 we describe a transmit diversity scheme for 2 transmit antennas that yields two-fold diversity by having pairs of users share two spreading codes. In Section 3 we outline some results regarding the extension of the scheme to the cases of more antennas on both sides of the link, complex input data and multipath channels. In Section 4 we present the results of a performance study that provides closed-form expressions for the uncoded BER in the presence of pilot-aided channel estimation. Numerical evaluations of the derived expressions are also included in Sections 3 and 4. Finally, Section 5 contains our conclusions.

2 Space-Time Spreading with 2 Transmit Antennas

Let the transmitted signals be

$$\begin{aligned} t_1 &= (1/\sqrt{2})(b_1\underline{c}_1 + b_2\underline{c}_2) \\ t_2 &= (1/\sqrt{2})(b_2\underline{c}_1 - b_1\underline{c}_2), \end{aligned} \tag{3}$$

on antennas 1 and 2, respectively. We call this transmit diversity scheme *Space-Time Spreading (STS)*, since each user's signal is spread across different antenna elements. Observe that we are using only two spreading codes, but are employing both codes with both data symbols. Notice also the normalization factor $1/\sqrt{2}$ which is used to equate the total transmit power to that of a single-antenna system. The received signals at the kth user terminal after despreading with $\underline{c}_1$ and $\underline{c}_2$ are

$$\begin{aligned} d_1 &= (1/\sqrt{2})(h_1b_1 + h_2b_2) + \underline{c}_1^\dagger \underline{n} \\ d_2 &= (1/\sqrt{2})(-h_2b_1 + h_1b_2) + \underline{c}_2^\dagger \underline{n}, \end{aligned} \tag{4}$$

where we have suppressed in the notation the dependence of the fading coefficients, the noise, and the despread signals on the particular receive terminal. (4) can be written as

$$\underline{d} = \frac{1}{\sqrt{2}} H \underline{b} + \underline{\nu}, \tag{5}$$

where

$$\underline{d} = \begin{bmatrix} d_1 \\ d_2 \end{bmatrix}, \qquad H = \begin{bmatrix} h_1 & h_2 \\ -h_2 & h_1 \end{bmatrix}, \qquad \underline{b} = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix}, \qquad \underline{\nu} = \begin{bmatrix} \underline{c}_1^{\dagger} \underline{n} \\ \underline{c}_2^{\dagger} \underline{n} \end{bmatrix}. \tag{6}$$

Multiplying $\underline{d}$ on the left by $H^{\dagger}$ and taking real parts we obtain

$$\mathrm{Re}\left\{H^{\dagger}\underline{d}\right\} \;=\; \tfrac{1}{\sqrt{2}} \begin{bmatrix} |h_1|^2 + |h_2|^2 & 0 \\ 0 & |h_1|^2 + |h_2|^2 \end{bmatrix} \underline{b} + \mathrm{Re}\left\{H^{\dagger}\underline{\nu}\right\}. \tag{7}$$

Denoting the kth column of H by $\underline{h}_k$, (7) can be written as

$$\mathrm{Re}\left\{\underline{h}_k^{\dagger}\underline{d}\right\} = (1/\sqrt{2})(|h_1|^2 + |h_2|^2) b_k + \mathrm{Re}\left\{\underline{h}_k^{\dagger}\underline{\nu}\right\}, \tag{8}$$

which has an SNR that possesses a two-fold diversity gain with respect to a single transmitter antenna. Note that two-fold diversity is achieved without having increased either the transmitted power or the number / length of spreading codes (1 code / user).

Adapting this two-antenna scheme to K users requires us to form $K/2$ pairs of users (K is assumed to be even), each of which is treated as above. Assuming that user k is paired with $k+1$, for $k = 1, 3, 5, \ldots, K-1$, the signals transmitted on antennas 1 and 2 are

$$\begin{aligned} t_1 \;&=\; \tfrac{1}{\sqrt{2}} \sum_{k=1}^{K/2} (b_{2k-1}\underline{c}_{2k-1} + b_{2k}\underline{c}_{2k}) = \frac{1}{\sqrt{2}} \sum_{k=1}^{K} b_k \underline{c}_k \\ t_2 \;&=\; \tfrac{1}{\sqrt{2}} \sum_{k=1}^{K/2} (b_{2k}\underline{c}_{2k-1} - b_{2k-1}\underline{c}_{2k}), \end{aligned} \tag{9}$$

respectively. Each mobile user despreads the incoming signal with the two sequences that it shares with its companion and processes the despread signals as already shown.

An alternative embodiment of STS can be obtained if instead of pairing different users we pair each user's so-called *sub-users*. In this case, b_1 and b_2 represent the even and odd stream of the data of the user of interest, respectively. This alternative approach provides the same diversity gains by using only one spreading code per user, but having it process symbols paired in time. Different choices of $\underline{c}_1$, $\underline{c}_2$ are then related to existing schemes. If $\underline{c}_1^T = [1 \;\; 0], \underline{c}_2^T = [0 \;\; 1]$, STS reduces to the space-time block code scheme proposed in [3] for TDMA. If, on the other hand, one chooses $\underline{c}_1^T = [\underline{c}_o \;\; 0_{1\times P}], \underline{c}_2^T = [0_{1\times P} \;\; \underline{c}_o]$, where $\underline{c}_o$ is any Walsh code of length P, this corresponds to the method proposed in [4] for the UMTS WCDMA system. Finally, with $\underline{c}_1^T = [\underline{c}_o \;\; \underline{c}_o], \underline{c}_2^T = [\underline{c}_o \;\; -\underline{c}_o]$, we construct the STS scheme that was recently proposed [5] (and adopted) in the cdma-2000 standard.

3 Generalization of STS to other cases

3.1 More than 1 receiver antennas

We first examine the effect of using more than one antenna at the mobile unit. It is foreseeable that a mobile may support more than one antenna as technology improves and wavelengths shorten. We assume that each receiver antenna is connected to the transmitter through an independent set of fading coefficients. This can be accomplished if the antennas are separated sufficiently far from one another. Let the signals given in (3) be transmitted as before. Then the processing shown in Section 2 may be performed at each of the N receiver antennas separately, yielding

$$\mathrm{Re}\,\{H_n^\dagger \underline{d}_n\} = (1/\sqrt{2})\left(|h_{1n}|^2 + |h_{2n}|^2\right)\underline{b} + \mathrm{Re}\,\{H_n^\dagger \underline{\nu}_n\}, \qquad n = 1,\ldots,N, \tag{10}$$

at the nth antenna. The N signals in (10) are now added, yielding

$$\sum_{n=1}^{N} \mathrm{Re}\,\{H_n^\dagger \underline{d}_n\} = \frac{1}{\sqrt{2}} \sum_{n=1}^{N} (|h_{1n}|^2 + |h_{2n}|^2)\underline{b} + \sum_{n=1}^{N} \mathrm{Re}\,\{H_n^\dagger \underline{\nu}_n\}. \tag{11}$$

The coefficient multiplying $\underline{b}$ in equation (11) is proportional to a χ^2_{2N} distribution. Moreover, its expected value is N. Thus, every receiver antenna boosts the expected SNR and contributes two degrees of freedom to the statistics of the received data. For simplicity, the remainder of the paper considers only one receiver antenna.

3.2 More than 2 transmitter antennas

We generalize the $M = 2$ transmitter antenna scheme by reexamining the transmitter and receiver structures presented in the previous section. We denote by S the $P \times 2$ matrix of transmitted signals across the two antennas. The pth row of S is the baseband representation of the signals on the two antennas at chip p. Then (3) implies that

$$S = C \cdot B, \tag{12}$$

where

$$C = [\,\underline{c}_1 \quad \underline{c}_2\,], \qquad B = \begin{bmatrix} b_1 & b_2 \\ b_2 & -b_1 \end{bmatrix}.$$

The baseband signal S is received at the mobile as

$$S\underline{h} + \underline{n} = CB\underline{h} + \underline{n},$$

which then despreads it by left-multiplying by $C^\dagger$, yielding

$$\begin{aligned} \begin{bmatrix} \underline{c}_1^\dagger \\ \underline{c}_2^\dagger \end{bmatrix} (S\underline{h} + \underline{n}) &= B\underline{h} + \begin{bmatrix} \underline{c}_1^\dagger \underline{n} \\ \underline{c}_2^\dagger \underline{n} \end{bmatrix} = \begin{bmatrix} b_1 & b_2 \\ b_2 & -b_1 \end{bmatrix} \begin{bmatrix} h_1 \\ h_2 \end{bmatrix} + \underline{\nu} \\ &= \begin{bmatrix} h_1 & h_2 \\ -h_2 & h_1 \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} + \underline{\nu} = H\underline{b} + \underline{\nu}, \end{aligned} \tag{13}$$

where H, $\underline{b}$, and $\underline{\nu}$ are defined in (6). The kth mobile then left-multiplies (13) by the kth row of $H^\dagger$. For M-fold diversity with M transmitter antennas, the key identity that allows each user to decode its symbol without decoding its neighbors is

$$\mathrm{Re}\,\{H^\dagger H\} = \underline{h}^\dagger \underline{h} \cdot I, \tag{14}$$

where $\underline{h} = [h_1 \ \cdots \ h_M]^T$.

In the most general case, we may attempt to group K' users at a time with L' spreading codes on M antennas, where $L' \geq K'$ and $L' \geq M$. With K users in the system, this would require a total of $L = (L'/K')K$ spreading codes. Ideally, we would like M-fold diversity with $L'/K' = 1$, which groups equal numbers of users and spreading codes, yielding $L = K$. This is accomplished in equations (12)–(13), where $L' = K' = M = 2$.

It is possible to show that (14) holds if the entries of the $L' \times K'$ matrix H depend linearly on $\underline{h}$ through real coefficients, and if

$$H^T H = \underline{h}^T \underline{h} \cdot I. \tag{15}$$

A general transmit diversity technique can be again described through (12). It is also possible to show that B matrices that obey $B^T B = \underline{b}^T \underline{b} \cdot I$ (where $\underline{b} = [b_1 \ \cdots \ b_{K'}]^T$) yield H matrices that obey (15). The transmitted signal to yield M-fold diversity is therefore as in (12), except that

$$C = [\, \underline{c}_1 \quad \underline{c}_2 \quad \cdots \quad \underline{c}_{L'} \,], \tag{16}$$

and B is an appropriately designed $L' \times M$ matrix of the input data.

To design M-fold diversity transmitter signals that have simple decoding algorithms, we may start with a design for B or H. Designs with H are more practical though since the mobile generally has the toughest constraints on processing power and complexity. In [3], *orthogonal designs* are used for multiple-antenna block coding schemes. For example, the matrices

$$H = \begin{bmatrix} h_1 & h_2 & h_3 & h_4 \\ -h_2 & h_1 & -h_4 & h_3 \\ -h_3 & h_4 & h_1 & -h_2 \\ -h_4 & -h_3 & h_2 & h_1 \end{bmatrix} \; ; \quad B = \begin{bmatrix} b_1 & b_2 & b_3 & b_4 \\ b_2 & -b_1 & b_4 & -b_3 \\ b_3 & -b_4 & -b_1 & b_2 \\ b_4 & b_3 & -b_2 & -b_1 \end{bmatrix}, \tag{17}$$

are an orthogonal design obeying $H^T H = \underline{h}^T \underline{h} \cdot I$, $B^T B = \underline{b}^T \underline{b} \cdot I$, with which we achieve an $L' = K' = M = 4$ diversity scheme. Notice that this scheme is non-rendunant since it achieves 4-branch diversity by spreading $K' = 4$ users with $L' = 4$ codes. However, such square designs for which $L' = K' = M$ exist only for $M = 2$, 4, or 8 [2]. We must be content for other values of M to have $L' = K' > M$.

3.3 Complex Data Symbols

We first consider the case $M = 2$. A square ($L' = K' = 2$) scheme that achieves full diversity in the case of complex symbols is

$$S = C \cdot B = [\, \underline{c}_1 \quad \underline{c}_2 \,] \cdot \begin{bmatrix} b_1 & b_2 \\ b_2^* & -b_1^* \end{bmatrix}. \tag{18}$$

As in the real-symbol case, the mobile despreads the received baseband signal as follows

$$C^\dagger (S\underline{h} + \underline{n}) = B\underline{h} + \begin{bmatrix} \underline{c}_1^\dagger \underline{n} \\ \underline{c}_2^\dagger \underline{n} \end{bmatrix},$$

but, unlike the real case, it has to conjugate the second entry of the result to yield

$$H\underline{b} + \begin{bmatrix} \underline{c}_1^\dagger \underline{n} \\ (\underline{c}_2^\dagger \underline{n})^* \end{bmatrix}.$$

The kth user now extracts its data symbol by multiplying on the left by the kth row of $H^\dagger$. Unlike the real-symbol case, both the real and imaginary parts of the result carry information. We have thus achieved $L' = K' = M = 2$.

In [3] it is shown that square matrices H obeying (15) for which $L' = K' = M' > 2$ do not exist, unlike the real-symbol case where they exist for $M = 2$, 4, and 8. They may, however, exist for $L' = K' > M > 2$, but we will not pursue further this issue here. We present instead a scheme for which $M = 3$, $K' = 3$, $L' = 4$: users are organized in groups of 3 with 4 spreading codes per group. The transmitted $L' \times M$ matrix B is

$$B = \begin{bmatrix} b_1 & -b_2 & -b_3 \\ b_2^* & b_1^* & 0 \\ b_3^* & 0 & b_1^* \\ 0 & -b_3^* & b_2^* \end{bmatrix},$$

and can be shown to provide full three-fold diversity at the handset.

3.4 Multipath channels

The preceding sections assume flat-fading. In the presence of multipath, the same transmission scheme can be used, but a RAKE-type architecture is required for reception as described in this section. Where favorable multipath conditions exist, a RAKE-type receiver can yield some additional diversity gains, beyond those attained by space-time spreading. For simplicity we assume $M = 2$ transmitter and $N = 1$ receiver antenna and BPSK inputs (generalizations to $M > 2$ and complex inputs are possible, as shown in the previous sections). We assume the channel from each transmitter antenna to the receiver antenna to comprise of J distinct paths, each of which experiences independent Rayleigh fading, expressed through the channel coefficient h_{mj}, $m = 1, 2$, $j = 1, \ldots, J$. Each pair of paths from the two transmitter antennas is assumed to arrive with the same set of delays to the receiver antenna and the path delays are approximately a few chips in duration, and small compared with the symbol period so that ISI can be neglected.

A spreading code $\underline{c}_l$ arrives at the receiver as a set of J signature vectors $\{\underline{c}_{l1}, \ldots, \underline{c}_{lJ}\}$, which are delayed versions of $\underline{c}_l$. As a result of multipath the received signal is now

$$\begin{aligned} \underline{r} &= \sqrt{\rho_1/2}\left(\sum_{j=1}^{J} h_{1j}(b_1\underline{c}_{1j} + b_2\underline{c}_{2j}) + \sum_{j=1}^{J} h_{2j}(b_2\underline{c}_{1j} - b_1\underline{c}_{2j})\right) + \underline{n} \\ &= \sqrt{\rho_1/2}\,(\mathbf{C}_1\mathbf{h}_1 - \mathbf{C}_2\mathbf{h}_2)\, b_1 + \sqrt{\rho_1/2}\,(\mathbf{C}_2\mathbf{h}_1 + \mathbf{C}_1\mathbf{h}_2)\, b_2 + \underline{n}, \end{aligned} \tag{19}$$

where $\mathbf{C}_l = \begin{bmatrix} \underline{c}_{l1} & \cdots & \underline{c}_{lJ} \end{bmatrix}$, $\mathbf{h}_l = \begin{bmatrix} h_{l1} & \cdots & h_{lJ} \end{bmatrix}^T$, $l \in \{1, 2\}$. The h_{mj} and the components of the receiver noise are normalized to be $\mathrm{CN}(0, 1)$, and ρ_1 represents the expected SNR of each multipath component (here assumed equal).

The sufficient statistics for detecting b_1 and b_2 are the following two vectors which are obtained by despreading $\underline{r}$ with $\mathbf{C}_1$ and $\mathbf{C}_2$ respectively, i.e.

$$\begin{aligned} \mathbf{d}_1 &= \mathbf{C}_1^\dagger\underline{r} = \sqrt{\rho_1/2}\,[\mathbf{R}_{11}\mathbf{h}_1 - \mathbf{R}_{12}\mathbf{h}_2]\, b_1 + \sqrt{\rho_1/2}\,[\mathbf{R}_{12}\mathbf{h}_1 + \mathbf{R}_{11}\mathbf{h}_2]\, b_2 + \mathbf{C}_1^\dagger\underline{n} \\ \mathbf{d}_2 &= \mathbf{C}_2^\dagger\underline{r} = \sqrt{\rho_1/2}\,[\mathbf{R}_{21}\mathbf{h}_1 - \mathbf{R}_{22}\mathbf{h}_2]\, b_1 + \sqrt{\rho_1/2}\,[\mathbf{R}_{22}\mathbf{h}_1 + \mathbf{R}_{21}\mathbf{h}_2]\, b_2 + \mathbf{C}_2^\dagger\underline{n}, \end{aligned} \tag{20}$$

where $\mathbf{R}_{ij} = \mathbf{C}_i^\dagger\mathbf{C}_j$. The extension of the detector (7) to the multipath case is the

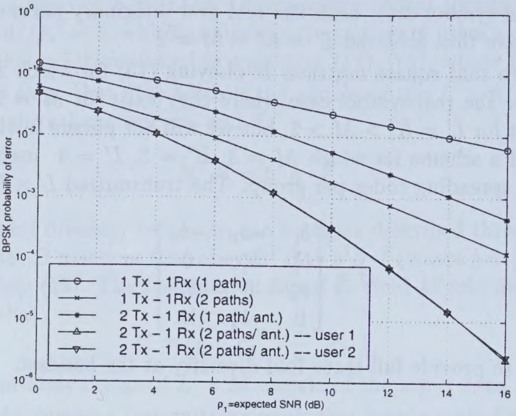


Figure 1: Probability of error for BPSK modulation with perfect channel knowledge

following (here we assume that perfect estimates of the fading coefficients are available)

$$
\begin{aligned}
\mathrm{Re}\left\{\mathbf{h}_1^{\dagger}\mathbf{d}_1 - \mathbf{h}_2^{\dagger}\mathbf{d}_2\right\} &\underset{b_1=-1}{\overset{b_1=1}{\gtrless}} 0 \\
\mathrm{Re}\left\{\mathbf{h}_2^{\dagger}\mathbf{d}_1 + \mathbf{h}_1^{\dagger}\mathbf{d}_2\right\} &\underset{b_2=-1}{\overset{b_2=1}{\gtrless}} 0
\end{aligned}
\tag{21}
$$

This equation shows a Rake-type receiver that generalizes (8) to the presence of multipath. It is also possible to show that the uncoded BER (conditioned on a particular channel realization) is given by

$$
\begin{aligned}
P_{e|\mathbf{H}}^{(1)} &= \frac{1}{2}Q\left[\mathrm{Re}\,(g_{11}+g_{12})\sqrt{\frac{\rho_1}{g_{11}}}\right] + \frac{1}{2}Q\left[\mathrm{Re}\,(g_{11}-g_{12})\sqrt{\frac{\rho_1}{g_{11}}}\right] \\
P_{e|\mathbf{H}}^{(2)} &= \frac{1}{2}Q\left[\mathrm{Re}\,(g_{22}+g_{21})\sqrt{\frac{\rho_1}{g_{22}}}\right] + \frac{1}{2}Q\left[\mathrm{Re}\,(g_{22}-g_{21})\sqrt{\frac{\rho_1}{g_{22}}}\right],
\end{aligned}
\tag{22}
$$

where G is the 2×2 matrix

$$
G = \begin{bmatrix} (\mathbf{C}_1\mathbf{h}_1 - \mathbf{C}_2\mathbf{h}_2)^{\dagger} \\ (\mathbf{C}_2\mathbf{h}_1 + \mathbf{C}_1\mathbf{h}_2)^{\dagger} \end{bmatrix} \begin{bmatrix} (\mathbf{C}_1\mathbf{h}_1 - \mathbf{C}_2\mathbf{h}_2) & (\mathbf{C}_2\mathbf{h}_1 + \mathbf{C}_1\mathbf{h}_2) \end{bmatrix}, \tag{23}
$$

and $Q(x) = 1/2\ \mathrm{erfc}(x/\sqrt{2})$. Due to the convexity of the Q function, we observe that there is always a penalty to be paid in bit error rate if g_{12}, g_{21} are nonzero.

In Figure 1 we plot the bit error performance of combinations of $M = 1, 2$ transmitter antennas and $J = 1, 2$ multipaths according to (22), where we have averaged over 10^5 independent Rayleigh channel realizations.

The codes $\underline{c}_1$ and $\underline{c}_2$ are orthogonal Walsh codes of length 128 and the delay between the two paths is 10 chips (corresponding to less than 10% of the symbol period, resulting in negligible ISI). The correlations between the delayed spreading codes are:

$$\mathbf{R}_{11} = \begin{bmatrix} 1 & +0.40625 \\ +0.40625 & 1 \end{bmatrix} \quad \mathbf{R}_{12} = \begin{bmatrix} 0 & -0.09375 \\ -0.09375 & 0 \end{bmatrix}$$

$$\mathbf{R}_{21} = \begin{bmatrix} 0 & -0.09375 \\ -0.09375 & 0 \end{bmatrix} \quad \mathbf{R}_{22} = \begin{bmatrix} 1 & -0.34375 \\ -0.34375 & 1 \end{bmatrix}.$$

We observe that the $M = 2$ system offers a substantial improvement over the single antenna system. The gain is almost 6 dB at the 10^{-2} uncoded BER level for the single-path case and of about 2 dB at the same level for the 2 path / antenna case. As expected, the highest gain is achieved when transitioning from a 1 to a 2-branch diversity system.

4 Effects of Channel Estimation Error

Decoding is predicated on knowledge of the fading coefficients for each of the M transmitter antennas that is provided to the mobiles through the transmission of M pilot signals. In this section we show that, under certain conditions, the decoding and channel estimation problems can be treated jointly, and that it is possible to obtain exact expressions for raw (uncoded) probability of error that directly incorporate the error in estimating the fading coefficients. Our expressions disclose two counteracting effects at work: increasing the number of transmitter antennas tends to give greater diversity gains, but if the total pilot power is fixed, the individual estimates for the fading coefficients deteriorate, and crosstalk increases among the users.

We first assume real-valued BPSK signals and that $M = L' = K'$, $N = 1$. After despreading, the message-bearing signal at any mobile is the L'-dimensional vector

$$\underline{y} = \left(\frac{\rho_1}{M}\right)^{1/2} B\underline{h} + \underline{\nu}, \tag{24}$$

and the pilot signal is the L'-dimensional vector

$$\underline{z} = \left(\frac{\rho_0}{M}\right)^{1/2} \underline{h} + \underline{\zeta}, \tag{25}$$

where ρ_1 and ρ_0 represent the expected SNR's for the message bearing and the pilot signals respectively at a particular mobile. The normalization by the square-root of M ensures again that the total transmitted power for each type of signal remains unchanged as the number of transmitter antennas changes. It is possible to show that the uncoded error probability is given by the following closed-form expression:

$$P_e = \frac{1}{(4+4\beta)^M} \sum_{j=0}^{M-1} \frac{\Gamma(2M-1-j)}{\Gamma(M)\Gamma(M-j)} \left[\frac{2\sqrt{4+4\beta}}{\sqrt{4\beta}+\sqrt{4+4\beta}}\right]^{1+j}, \tag{26}$$

where $\beta = \frac{\rho_0\rho_1}{M(M+\rho_0+M\rho_1)}$ and $\Gamma(n) = (n-1)!$. The probability of error (26) depends only on M and β. The effect of imperfect knowledge of the fading coefficients, as manifested by a finite value for the expected SNR of the pilot ρ_0, is an effective reduction in the SNR of the message bearing signal ρ_1. We can define an effective SNR to be the value of ρ_1, that, with perfect knowledge of $\underline{h}$ (e.g., $\rho_0 = \infty$) yields the same β evaluated for finite ρ_0. When $\rho_0 = \infty$, we have $\beta = \rho_1/M$. Hence, we solve

$$\frac{\rho_{\text{eff}}}{M} = \frac{\rho_0\rho_1}{M(M+\rho_0+M\rho_1)} \quad \rightarrow \quad \rho_{\text{eff}} = \frac{\rho_1}{1+M(1+\rho_1)/\rho_0}.$$

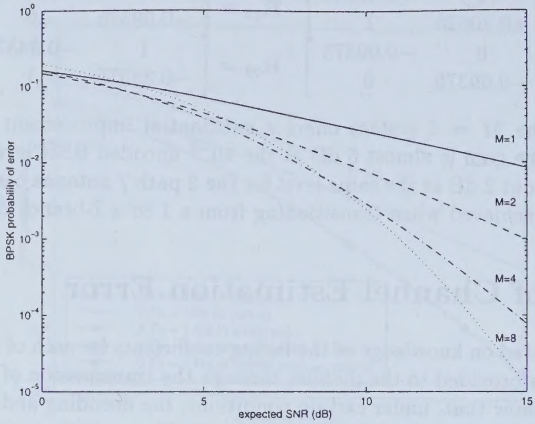


Figure 2: Probability of error for BPSK modulation with pilot-aided channel estimation

Holding the total transmitter power fixed while increasing the number of transmitter antennas trades off two opposing effects: gain in diversity versus reduction in effective SNR. The gain in diversity is due to the M independent paths between the transmitter and receiver; the reduction in effective SNR is due to the crosstalk caused by the lack of knowledge of the M propagation coefficients.

Figure 2 shows plots of the probability of error according to (26) for $M = 1, 2, 4, 8$, as functions of ρ_1, where $\rho_0 = 10\rho_1$ (i.e., 10 dB greater). For $\rho_1 = 5$ dB, using two antennas rather than one yields approximately a 2 dB gain. Comparing Figures 1 and 2, there is little penalty incurred by estimating the fading coefficients from noisy pilot signals. Notice also that for sufficiently low SNR's, increasing the number of antennas can actually increase the probability of error.

We can extend the above result to complex-valued signals with $|b_k| = 1$. Now $M = K' = L'$ is only possible for $M = 1$ or 2. Following the same steps as for the real case, we obtain the exact pairwise probability of error,

$$P_e = \frac{1}{(4+\delta^2\beta)^M}\sum_{j=0}^{M-1}\frac{\Gamma(2M-1-j)}{\Gamma(M)\Gamma(M-j)}\left[\frac{2\sqrt{4+\delta^2\beta}}{\sqrt{\delta^2\beta}+\sqrt{4+\delta^2\beta}}\right]^{1+j}, \tag{27}$$

where δ is the Euclidean distance between the two QPSK signals. Note that this expression is exactly equivalent to the BER expression for the real case (26) where $\delta = 2$. Figure 3 shows plots of the pairwise probability of error according to (27) for $M = 1, 2$ as functions of ρ_1, where again $\rho_0 = 10\rho_1$. In this case $\delta = \sqrt{2}$, corresponding to two QPSK signals that differ by 90 degrees in phase.

5 Conclusions

We have devised a novel space-time spreading scheme for the forward link of CDMA systems that provides diversity gains without requiring the transmitter to estimate the

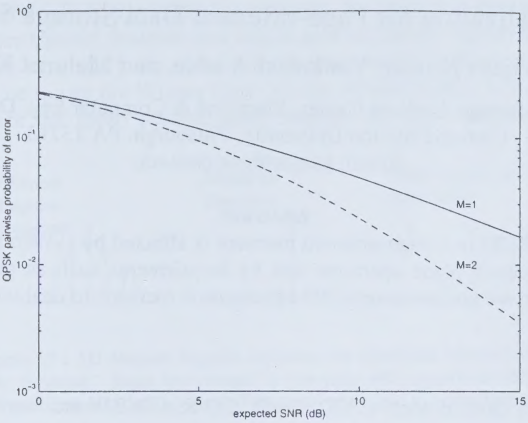


Figure 3: Pairwise probability of error for QPSK modulation with pilot-aided channel estimation

channel propagation coefficients, and with little extra system complexity. The essence of the scheme is for groups of users (or sub-users) to share common sets of spreading codes. Where favorable multipath conditions exist, a RAKE-type receiver can yield some additional diversity gains, beyond those attained by space-time spreading.

Adding extra antennas requires the addition of extra pilot signals to enable the mobiles to estimate the multiple-antenna propagation coefficients. We have shown, however, that even with the increased channel estimation error that this entails, significant net diversity gains can be realized.

Other possible benefits of our scheme, that remain to be quantified, are the mitigation of both the power control problem as well as the inter-cell interference problem. Increased diversity implies that adjustments to power do not have to occur as frequently. Dividing the downlink transmission for each mobile among two or more spreading codes should make the interference appear more Gaussian and more white in other cells.

References

[1] B. Hochwald, T. Marzetta, C. Papadias, “A transmitter diversity scheme for CDMA systems based on space-time spreading,” Bell Labs Technical Memorandum, May 1999.

[2] A. V. Geramita and J. Seberry, *Orthogonal designs: quadratic forms and Hadamard matrices*, New York: Marcel Dekker, 1979.

[3] V. Tarokh, H. Jafarkani, and A. R. Calderbank, “Space-time block codes from orthogonal designs,” *IEEE Trans. Information Theory*, July 1999.

[4] Texas Instruments, “Space-time block coded transmit antenna diversity for WCDMA,” proposed TDOC# 662/98 to ETSI SMG2 UMTS standards, December 1998.

[5] Lucent Technologies, “Downlink diversity improvements through space-time spreading,” proposal 3GPP2-C30-19990817-014 to the cdma-2000 standard, August 1999.

2D Equalization for Page-oriented Data Storage Systems

B.V.K. Vijaya Kumar, Venkatesh Vadde, and Mehmet Keskinoz

Data Storage Systems Center, Electrical & Computer Eng. Dept.,
Carnegie Mellon University, Pittsburgh, PA 15213
Email: kumar@ece.cmu.edu

Abstract

The bit error rate (BER) in a page-oriented memory is affected by system parameters such as contrast ratio, frequency plane apertures and by impairments such as pixel spreading and noise. In this paper, we present several 2D equalization methods to deal with these problems.

1. Introduction

Optical data storage devices such as CD, DVD, CD-R, CD-RW etc. have become essential components in data warehousing. Traditionally, optical storage has involved writing marks on a disk with each mark representing a single bit. For a given wavelength λ, the mark-size is diffraction limited to the order of λ^2 [1-2]. This intrinsically limits the achievable areal storage density and hence capacity. In page-oriented optical storage, it is possible to significantly extend the capacity by using the third dimension leading to volumetric storage. Instead of data-marks in 2D, we now write data-pages (i.e., pixellated patterns for 0s and 1s) in small 3D storage volumes, with each page holding about 1 Mb at a time [3]. Furthermore, it is possible to multiplex many data-pages within the same storage volume. This enables us to target storage capacities of a few 100 GB per disk (assuming a 0.5-1 mm thickness), taking us into the next generation of optical storage.

Several strategies can be employed to increase the density in page-oriented optical storage, such as increasing the pixel count per page, multiplexing more pages per storage volume, or packing more storage volumes in the medium. For a fixed pixel size, the field of view of the data page imposes imaging constraints limiting the number of pixels per page. As we increase the number of multiplexed pages per unit volume, the readout diffraction efficiency (or signal strength) decreases resulting in low signal-to-noise ratios (SNR). The last approach i.e., using closely packed storage volumes in the medium implies that we have to use small focal-plane apertures to eliminate cross talk between them. Smaller focal plane (i.e., frequency plane) apertures tend to bandlimit the system and thereby introduce intersymbol interference (ISI). In practice, one is often faced with severe ISI under conditions of low SNR. Under such conditions, it becomes imperative to use advanced equalization and detection to retrieve the read-back data. In this paper, we will describe the use of several equalization and detection algorithms devised for a 2D page-oriented storage channel such as in holographic storage or in 2-photon optical storage.

2. Equalization and Detection Techniques

The goal of equalization and detection is to minimize the effects of ISI without degrading SNR. The task of minimizing ISI can be approached in several ways. The challenge is to achieve ISI reduction without letting noise amplification or error propagation effects to degrade the SNR. We will start with the simplest approach, namely deconvolution.

In this equalization (also known as zero forcing equalization), one tries to fully eliminate the ISI by equalizing the discrete channel impulse response to a delta function. Fig.1 shows a sample stem plot of a 1-D channel impulse response, the equalized channel impulse response and finally the filter transfer function that would accomplish this equalization. Clearly, such a filter would tend to amplify the noise at high frequencies making it less attractive. An alternative would be to use the Wiener filter [4] that differs from a zero-forcing filter in that it tries to minimize the mean squared error between the ideal outputs and the equalized outputs.

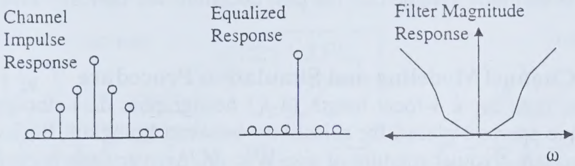


Fig.1: Example of a 1D channel impulse response, the equalized channel response and the filter transfer function. Since the channel is low-pass with significant ISI, the filter that inverts the channel tends to be high-pass. This could often lead to noise-enhancement.

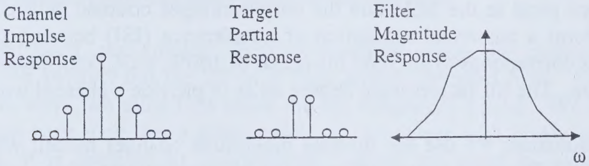


Fig.2: Hypothetical channel impulse response and (1+*D*) type of target response along with equalizer transfer function. While the (1+*D*) PR target makes ISI manageable, the problem of noise enhancement found in ZF equalizers can be avoided.

More advanced methods of equalization attempt to control the ISI by filtering. Partial response (PR) equalization is one such method where we attempt to equalize the channel impulse response to a short extent target [5-6]. This enables the partial response equalizer to avoid the noise enhancement seen in other methods. However, the post-equalization data now has remnant ISI. The optimum approach to dealing with this remnant ISI involves maximum likelihood sequence detection (MLSD). Using an appropriate short-extent target for equalizing the channel impulse response can also reduce the MLSD complexity. In Fig.2, we show a hypothetical 1D channel impulse response and a simple (1+*D*) type of target response, where *D* stands for unit delay. We also show the required equalizer transfer function. Clearly, this equalizer is low-pass in nature and thus avoids the noise enhancement.

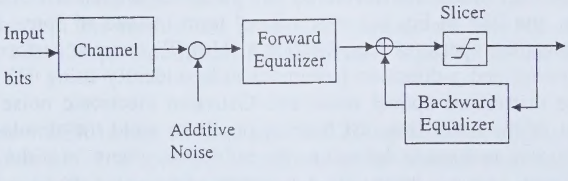


Fig.3: Block diagram of a DFE. The forward equalizer makes the ISI causal, removing all ISI from future (undetected) bits. The backward equalizer ensures that the remaining ISI from already detected bits is subtracted off.

Another class of equalizers, known as decision feedback equalization (DFE) relies on the notion of decision feedback. These methods use already made decisions to subtract the ISI from the already detected bits. In DFE, the goal of the forward equalizer is to make the ISI entirely causal [7] i.e., the trailing ISI bits are equalized to zero. Thus, the only ISI left is from already detected bits and that can be subtracted off.

If some of the leading ISI in DFE is left unsubtracted, one could use a tree-search approach often referred as the fixed-delay tree search (FDTS) algorithm [8]. The drawback with DF-based methods is that they assume that the past decisions are correct. This can lead to error propagation.

3. Review of Channel Modeling and Simulation Procedure

In this paper we consider a 4-focal length (4-f_L) holographic data storage system (HDSS) with a focal plane aperture placed for selectivity between hologram stacks [9]. It has been shown that the square Nyquist aperture of size W = $\lambda f_L/\Delta x_1$ very nearly optimizes the storage density [10,11], where λ is the wavelength and Δx_1 refers to the spatial light modulator's (SLM) pixel pitch. We therefore focus on a Nyquist aperture in the frequency plane. We assume that the amplitude contrast ratio of the SLM is 3 for binary and 5 for grayscale storage. In order to simulate ISI, we consider the standard 12 nearest neighbors [12]. We oversample each pixel in the SLM and the output charges coupled device (CCD) 100x100 times and perform a numerical simulation of interference (ISI) between pixels. The data simulated by us corresponds to an SLM fill-factor of 100%, a CCD fill-factor of 40% and the Nyquist aperture. The fill-factors were chosen so as to provide a channel with good linearity.

For linear equalization, we use the discrete magnitude channel model, where the positive square roots of CCD output values are modeled as linear superposition values of the input data. We note that the magnitude channel model leads to a discrete PSF as given below:

$$h_M[k,l] = \int_{(k-\alpha_2/2)\Delta x_2}^{(k+\alpha_2/2)\Delta x_2} dx \int_{(l-\alpha_2/2)\Delta y_2}^{(l+\alpha_2/2)\Delta y_2} \left(\left\{ \mathrm{rect}\left(\frac{x}{\delta x_1}, \frac{y}{\delta y_1}\right) ** \mathrm{sinc}\left(\frac{xd}{\lambda f_L}, \frac{yd}{\lambda f_L}\right) \right\} \right) dy \qquad (1)$$

where δx_1, δy_1 are SLM pixel dimensions; Δx_2,Δy_2 are CCD pixel pitches, α_2 is the linear fill factor of the CCD pixels, d is the size of the rectangular aperture in the frequency plane and ** denotes 2-D convolution operation. If r, a, h_M and n_M denote the CCD output, SLM input, the discrete channel PSF and noise sample respectively, the linear model can be expressed as

$$\sqrt{r} = a ** h_M + n_M \qquad (2)$$

The only other channel model considered by us (for ZF equalization alone) is based on an intensity PSF i.e., the PSF in Eq. (1) with a *Sinc*^{2} term instead of *Sinc*. Since the PSF is separable for rectangular apertures (i.e., $h[k,l] = h_x[k]\, h_y[l]$), we can perform equalization in the x-direction (rows) and y-direction (columns) independently using 1D designs. In our simulations, Rice distributed optical noise and Gaussian electronic noise is added to the magnitude values of the noise free, ISI bearing pixels to yield the simulated CCD output. The SNR used for our analysis is defined as the ratio m/σ, where, m is the mean of the ON (brightest level) pixels and σ is the standard deviation of the noise affecting the ON pixels.

4. Extension of Equalization/Detection to 2D Data

For holographic storage, equalization/detection techniques must address the 2D nature of data. Specifically, in a 2D channel with ISI in both x (row) and y (column) directions, one must decouple the different rows (or columns) before applying the 1D techniques to the rows (or columns). Towards that end, we pursue two approaches. In the first approach, we completely eliminate ISI along columns using a ZF equalizer. The rows within a page can then be independently decoded.

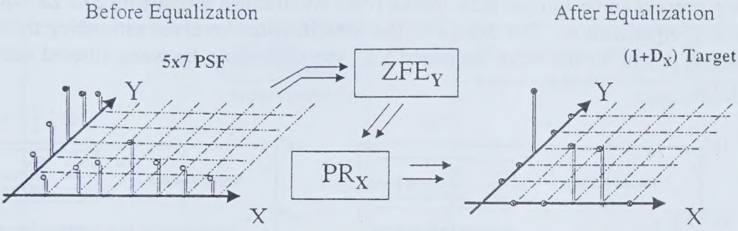


Fig.4. The 2-d 5x7 PSF is shown pictorially before and after equalization through its components along the X and Y-directions. ZF equalization along Y helps to entirely eliminate ISI in Y while PR equalization along X reduces the extent of ISI to two adjacent pixels.

In the second approach, we employ decision feedback along the columns i.e., before detecting a row, we subtract out the ISI in it from already detected rows. For such a DF-based approach to work, we must first make the ISI temporally causal. We ensure this by equalizing the data page such that a given pixel only interferes with pixels to the right and above it. Thus, it is possible to independently detect the rows by starting from the bottommost row and proceeding upwards. We employ the ZF and DF-based approaches to adapt all our advanced techniques namely, PRML, DFE and FDTS. The resulting techniques will be referred to by the appropriate prefix such as, ZF-PRML, DF-PRML, ZF-DFE, DF-DFE, ZF-FDTS and DF-FDTS. In the following subsections we will present simulation results for ZFE, MMSE, PRML and DF-based equalization/detection algorithms.

5. Results and Discussion

5.1 Linear Equalization

For simulation results, we have noted in the past [9] that when the channel is optical noise dominated, the magnitude channel provides better linearity, especially at moderate to low CCD fill-factors. Therefore for the optical noise dominated case, we pick a channel with 100% SLM fill-factor and 40% CCD fill-factor. The other case considered by us is an electronic noise dominated channel. The intensity channel model appears to be better in this case, especially at high CCD fill-factors. Therefore, we illustrate this type of a channel with simulations having a 100% SLM fill-factor and 80% CCD fill-factor. All measurements (i.e., each BER data point) in this section and subsequent sections are based on simulation of 10 pages, each with 128x128 pixels. Thus, BER estimates equal to 10^{-5} or larger are fairly reliable.

In Fig.5 (a) we show BER results for 3x3 and 5x5 ZF equalization obtained for an optical noise dominated channel. The aperture assumed is of the Nyquist size. Magnitude channel model performs better in this case. The BER versus the SNR shows an improvement from a 3x3 equalizer to a 5x5 equalizer. The improvement factor appears to be ~2-3. This can be attributed to the greater extent of ISI from a magnitude channel model. The equalization gain possible in going from 3x3 to 5x5 filters appears to be smaller at higher densities (i.e., lower SNRs) and as such may not justify the additional complexity incurred in a 5x5 filter.

Fig.5 (b) shows the BER before and after equalization for a channel with dominant electronic noise. The BER with 3x3 and 5x5 filtering obtained using the better-performing intensity channel appear to show only a marginal BER improvement. So, 3x3 equalization kernel size appears to be sufficient under such conditions.

Next, we present some sample BER trends from simulations employing 3x3 ZF equalization and MMSE equalization. The design of the MMSE-filter involves estimating the noise and minimizing power in the error sequence i.e., the difference between filtered samples and ideal signal.

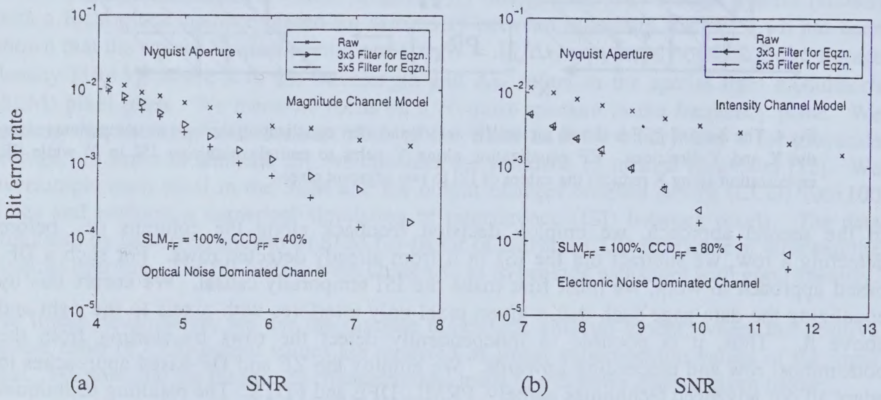


Fig.5 (a) & (b) BER results before and after equalization for the Nyquist aperture channel with dominant optical noise (*left*) and for dominant electronic noise case (*right*). The magnitude channel model is employed for the low CCD fill-factor, optical noise dominated case in Fig.5 (a) while the intensity model is employed to equalize the high CCD fill-factor channel with dominant electronic noise (Fig.5 (b)).

BER results with MMSE for an optical noise and electronic noise dominated channel are shown in Fig.6 (a) and Fig.6 (b), respectively. The optical noise dominated channel with relatively low fill-factor shows a noticeable but not significant BER gain by employing the MMSE equalizer instead of the ZF equalizer. The same gain obtained in an electronic noise dominated channel with the 3x3 MMSE filter (now employing a higher CCD fill-factor of 80% with the intensity channel model) is considerably higher. We believe this is due to the lower noise enhancement when the intensity model is employed for equalization.

5.2 **PRML**

For PRML, the simulated CCD outputs were equalized and then detected using the Viterbi algorithm. For PR equalization, we first need to choose a PR target that closely matches the channel frequency response to avoid noise enhancement by the equalizer. We find (by trial and error) that for the parameters chosen by us, the 1D PR target that best matches the channel is $(1+\alpha D)$, where $\alpha \cong 0.3$ for the Nyquist aperture and $\alpha \cong 1$ for sub-Nyquist aperture, where D is the unit delay operator. There are four types of channels that we consider here resulting from two apertures – Nyquist (for grayscale storage) and sub-Nyquist aperture (for binary storage), and two types of noise – dominant optical noise and dominant electronic noise. For each type of channel, we simulated about 10 pages, each of size 128x128 pixels, and averaged the BER obtained over the 10 pages. It is of interest to compare the performance and density gains obtained through PRML with respect to the unequalized Nyquist aperture case. To this end, we include the BER obtained for the unequalized Nyquist aperture case in our sub-Nyquist aperture plots, and designate the corresponding curve as the "baseline" case.

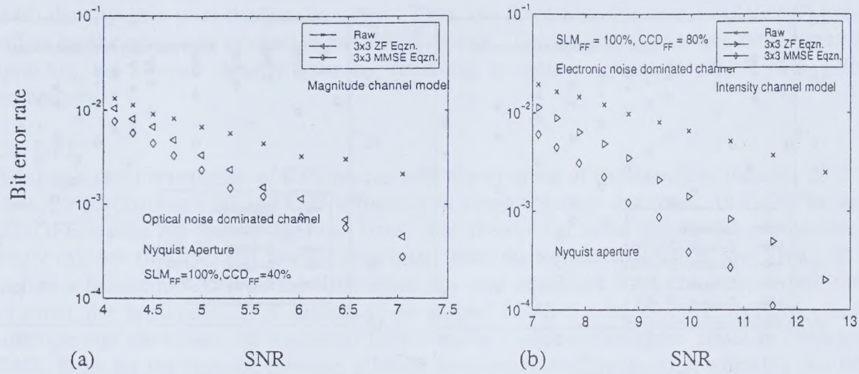


Fig.6 (a) & (b) BER results with 3x3 ZF and MMSE equalization for the Nyquist aperture channel with dominant optical noise (*left*) and for dominant electronic noise case (*right*). The magnitude channel model employed for the low CCD fill-factor, is more low-pass (than intensity channel), leading to a lesser BER gain from ZFE to MMSE (compare (a) and (b)).

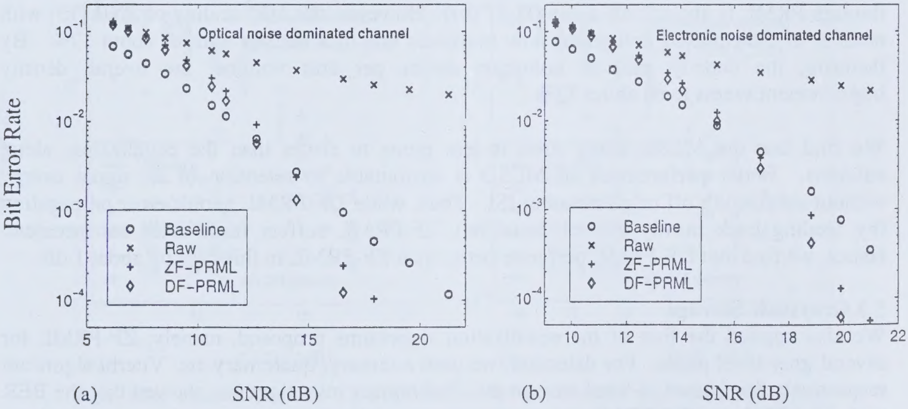


Fig.7 BERs before and after equalization using PRML for the sub-Nyquist aperture for (a) an optical noise dominated channel as well as (b) an electronic noise dominated channel. Baseline refers to the case of the Nyquist aperture without equalization.

Since PRML has the potential to extract higher densities from sub-Nyquist apertures, we will show only the sub-Nyquist results for binary storage. Fig.7 shows the BER vs. SNR obtained using ZF-PRML and DF-PRML. Several interesting observations can be made from Fig.7. The sub-Nyquist aperture can never attain a BER goal of 10^{-3} without equalization. For the target BER 10^{-3}, we find that PRML can yield an SNR improvement of 2 dB with the sub-Nyquist aperture for dominant optical noise. Using the $1/N$ scaling laws [9] we obtain an increase in hologram density per stack of $10^{0.1} \cong 1.26$ or about 26%. We note that the sub-Nyquist aperture chosen is physically smaller than the Nyquist aperture by a factor of 0.85, implying the overall density gain with an OND channel is upto 48%.

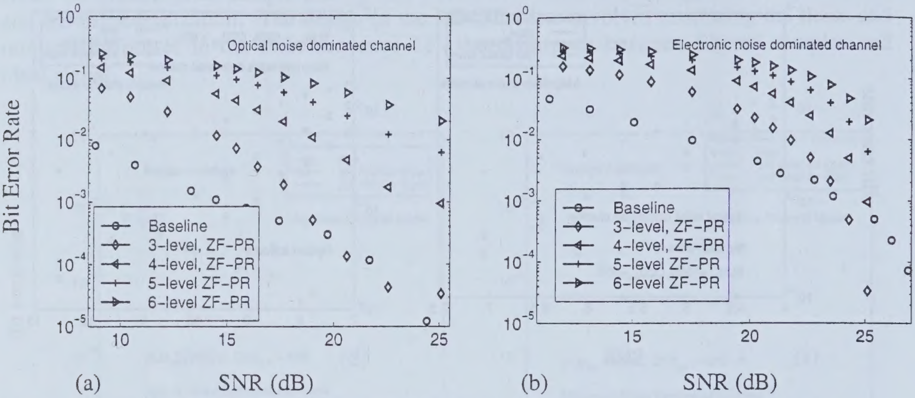


Fig.8 We consider different levels of grayscale storage (3-6 levels) and plot the equalized BER along with the baseline case as a function of SNR for **(a)** an optical noise dominated channel and **(b)** an electronic noise case. Both cases represent the Nyquist aperture here.

For the electronic noise dominated channel with sub-Nyquist aperture, the SNR improvement through PRML is about 2 dB again (Fig.7 (b)). However, the $1/N^2$ scaling of SNR [13] with number of multiplexed holograms now translates this to a density gain of about 12%. By factoring the density gain in hologram stacks per unit volume, the overall density improvement seems to be about 32%.

We find that the MLSD along rows is less prone to errors than the equalization along columns. Better performance of MLSD is attributable to retention of all signal energy without subtracting off or eliminating ISI. Thus, while DF-PRML avoids error propagation (by feeding back mostly correct decisions), ZF-PRML suffers from noise enhancement. Hence, we find that DF-PRML performs better than ZF-PRML in this case by about 1 dB.

5.3 Grayscale Storage

We also applied the first of the equalization algorithms proposed, namely, ZF-PRML for several gray-level pages. For detection, we used a ternary, quaternary etc. Viterbi algorithm respectively for 3-level, 4-level storage etc. Preliminary investigations showed that the BER with ZF-PRML and DF-PRML are not significantly different. Hence, we have restricted our study for multiple grayscale levels to ZF-PRML algorithm. The BER results with the ZF-PRML algorithm are shown in Fig.8 (a) and Fig.8 (b) for the optical noise dominated and electronic noise dominated cases, respectively. The baseline case here for comparison is again 2-level, Nyquist aperture case with no equalization. From Fig. 8 (a), we find that a target BER of 10^{-3} implies that we cannot use more than 4 gray-levels. With 4-level storage, we find that the SNR required for a BER of 10^{-3} is about 25 dB. This represents a 10.5 dB loss with respect to the baseline case. The enhancement in density due to the extra information per pixel is not enough to counter this huge SNR loss. The SNR loss accompanied with the diminishing separation between output levels far outweighs the additional storage density resulting from the extra levels. Clearly, there is no incentive to store 4-level grayscale pages for an optical noise dominated channel. From the electronic noise case in Fig.8 (b), we notice that the 4-level case leads to only a 1.5 dB SNR loss. Indeed, this translates to an overall density gain that is impressive compared to the baseline case. The net change in density can be shown to be = $10^{(-1.5/40)} \times 2 = 0.92 \times 2 = 1.84$, or an 84% density gain over the baseline case. Thus, the SNR loss due to multiple levels can be offset by the grayscale advantage and PRML gains. The results shown here only attempt to quantify the overall density gain by choosing to detect grayscale pages with PRML techniques.

5.4 DFE

We begin our presentation of DFE results with the simplest of the detectors, namely, ZF-DFE detector performing ZFE and DFE respectively, along columns and rows. In Fig.9, we show ZF-DFE results for various aperture sizes. For the optical noise dominated case shown in Fig.9 (a), we find that the raw BER is better than the equalized BER at low SNR. If we define a breakpoint SNR as the SNR when raw and equalized BER coincide, for the OND channel, the breakpoint SNR appears to be around 16-dB for the Nyquist aperture. As the aperture size decreases, the equalized BER steadily rises increasing the effective breakpoint SNR. Even for the Nyquist aperture, a 16-dB breakpoint SNR is too high when the raw BER is already 2×10^{-4}. For the electronic noise dominated channel, the trend is identical with the SNR breakpoint occurring at around 17 dB for the Nyquist aperture. At smaller apertures the breakpoint SNR again increases. As such, it appears for both the optical and electronic noise channels that the benefit of DFE is not appreciable, particularly at low SNR.

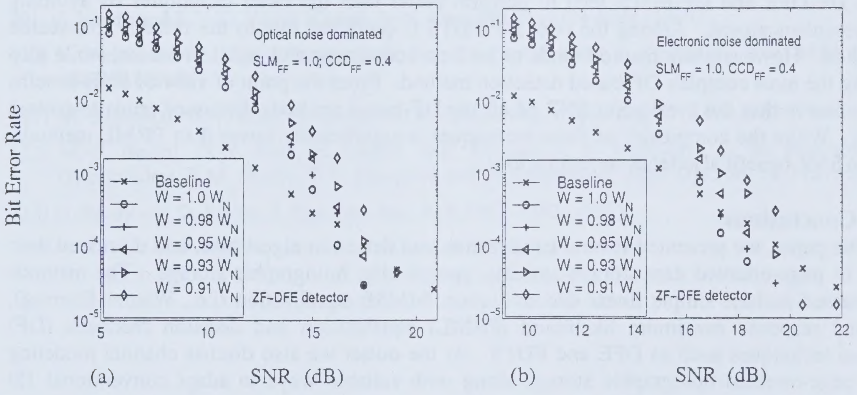


Fig.9 BER obtained with the ZF-DFE detector for the Nyquist and several sub-Nyquist apertures, as a function of SNR. An optical noise dominated channel (*left*) and an electronic noise dominated channel (*right*) are shown here. The baseline case refers as usual to the raw BER results (i.e., without any equalization) from the Nyquist aperture.

The BER for different detection algorithms have been plotted for the Nyquist aperture in Figs.10. The ZF-DFE detector is least effective in reducing BER, followed by the ZF-FDTS detector, the DF-DFE detector and the DF-FDTS detector respectively. The order of performance can be explained from noise enhancement and error propagation considerations as seen before. To summarize, the ZF-DFE suffers noise enhancement along *y* and employs an error propagation prone technique along x. The ZF-FDTS detector does slightly better owing to the tree-search based detection along x although it still suffers from noise enhancement.

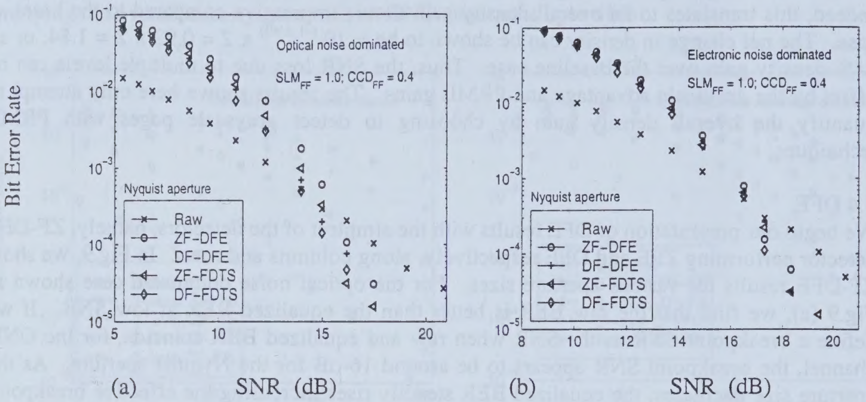


Fig.10 The raw (unequalized) and equalized BER shown as a function of the SNR for the Nyquist aperture for different detectors employed namely, ZF-DFE, DF-DFE, ZF-FDS and DF-FDTS. An optical noise dominated case (*left*) and an electronic noise dominated channel (*right*) are considered. The feedback filter length for the case of DFE was assumed M=7, and for FDTS-based algorithms, the depth of tree-search was fixed at τ=3.

The DF-DFE and DF-FDTS tend to perform better than the other techniques by avoiding noise-enhancement. Among the two, DF-FDTS is preferred due to the reliable tree-search method. However, this method tends to be time-consuming and serial in nature, while also being the most complex DF-based detection method. From the point of view of SNR benefit, we observe that the breakpoint SNR of all the DF-based methods discussed tends to remain high. While the complexity of these techniques is significantly lower than PRML methods, their SNR benefit also tends to remain low.

6. Conclusions

In this paper, we presented several equalization and detection algorithms, and discussed their use in page-oriented data storage systems, specifically, holographic storage. The methods discussed include simple linear deconvolution, MMSE equalization (i.e., Wiener filtering), partial response maximum likelihood (PRML) equalization and decision feedback (DF) based techniques such as DFE and FDTS. At the outset we also discuss channel modeling for page-oriented holographic storage along with suitable ways to adapt conventional 1D equalization techniques to 2D page-oriented systems.

We found that simple 3x3 deconvolution could provide upto an order of magnitude BER improvement when used with the Nyquist aperture. 5x5-filtering appeared to provide additional benefit only with the magnitude channel model. Simulation results also showed that MMSE equalization could extract a factor of 2 or so additional BER gain over ZFE, especially with the intensity model. DF-based approaches appeared unsuitable to holographic storage in their conventional form. The poor performance of DFE appears to be due to the low noise-margins available in the slicer or tree-search detector. With suitable PR equalization, we found that one could use sub-Nyquist apertures gaining upto 48% in density over the unequalized Nyquist aperture ("baseline") case. Multiple-level grayscale storage with PR equalization appeared to give upto 84% density gain over the baseline case, for an electronic noise dominated system. In this paper, we detected the grayscale pages employing ternary, quaternary etc. Viterbi detectors. Overall, it appears that PR equalization is most suitable to holographic storage systems yielding impressive density gains over the unequalized case.

Acknowledgments
This research conducted at the Data Storage Systems Center of Carnegie Mellon University was supported in part by the National Science Foundation under grant #ECD-8907068 and in part by DARPA and the Holographic Data Storage System (HDSS) consortium.

References

[1] M. Mansuripur and G. Sincerbox, Proc. IEEE, **85**, 11, p. 1780-96 (1997)

[2] P. Asthana, B.I. Finkelstein, and A.A. Fennema, IBM J. Res. & Dev., **40**, 5, 543-58 (1996)

[3] J.H. Hong and D. Psaltis, Laser Focus World, **32**, 4, p. 119-22 (1996)

[4] D. E. Lee and D.G. Messerschmitt, "*Digital Communications,*" Kluwer Academic, 2nd ed. (1993)

[5] Kabal and Pasupathy, IEEE Trans. Comm., **COM-23**, 9, 921 (1975)

[6] H.K. Thapar and A.M. Patel, IEEE Trans. Mag., **MAG-23**, 3666 (1987)

[7] J. M. Cioffi, G. P. Dudevoir, M. V. Eyuboglu, and G.D. Forney Jr., IEEE Trans. Comm., **43**, 10, 2582-2594 (1995)

[8] J.J. Moon, L.R. Carley, IEEE Trans. on Magn., **26**, 6, 3155 (1990)

[9] V. Vadde and B.V.K. Vijaya Kumar, Appl. Opt., **38**, 20, 4374-4386 (1999)

[10] V. Vadde, B.V.K. Vijaya Kumar, G.W. Burr, H. Coufal, and J. A. Hoffnagle, Proc. SPIE, **3401**, 194-200 (1998)

[11] M.-P. Bernal, G.W. Burr, H. Coufal and M. Quintanilla, Appl. Opt., **37**, 5377-5385 (1998)

[12] M.-P. Bernal, G.W. Burr, H. Coufal, R.K. Grygier, J. A. Hoffnagle, C. M. Jefferson, E. Osterschultze, R.M. Shelby, G.T. Sincerbox and M. Quintanilla, Appl. Opt., **36**, 14, 3107 (1997)

[13] D. Brady and D. Psaltis, J. Opt. Soc. Am. A, **9**,1167–1182 (1992).

Towards Soft Output APP Decoding for Nonsystematic Nonlinear Block Codes

Kofi D. Anim-Appiah and Steven W. McLaughlin

The School of Electrical and Computer Engineering
Georgia Institute of Technology, Atlanta, GA 30332

`kofi,swm@ece.gatech.edu`

Abstract

We discuss some aspects of obtaining log *a posteriori* probability ratios for data bits that are encoded by a nonsystematic, nonlinear block modulation code. This problem may arise in some system configurations for magnetic data storage channels where a constraint demodulator receives soft channel information and must in turn, provide soft information to a soft-input error control code decoder.

1 Introduction

Interest in the use of interleaved parallel ("turbo") [2] and serial concatenated codes on magnetic data storage channels continues to grow. But even as this prospect materializes, there still remain some aspects of the interface between the requirements of these input-constrained channels and those of soft-input *a posteriori* probability (APP) detectors that have not been fully studied. One of these aspects arises directly from the use of a constraint (or modulation) code to satisfy the input runlength requirements of the recording channel; these runlength requirements are often based on the family of (d, k) constraints which require that any successive pair of ones be separated by at least d and at most k zeros [5]. In a conventional recording system, the arrangement of the modulation and error control code (ECC) is characterized by the outer ECC/inner modulation code as shown in Fig. 1.

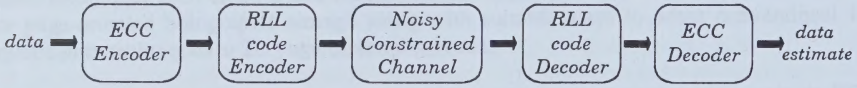


Figure 1. Conventional RLL/Error Correction coding arrangement

When the ECC has a decoder with soft-input requirements, as is the case with a turbo code, this arrangement appears to have some difficulty. The modulation code is often implemented with a nonsystematic block code which is also nonlinear; it is generally not known how to obtain log APP ratios for the constraint encoder input bits for such a generic code. This is primarily due to the lack of a sufficiently simple description of the encoder (i.e., a trellis) which employs a large number of relatively long code words. In order to minimize the rate loss of the constraint code (which typically has no distance properties), it's rate tends to be high; typical constraint code rates are of the form $(n-1)/n$ with n perhaps 9 or greater.

Several alternative system configurations that circumvent this difficulty have been recently proposed — Anim-Appiah & McLaughlin report in [1] on a configuration that is characterized by an outer constraint code/inner turbo code, while Fan & Cioffi [4] describe methods of adapting the original configuration of Bliss to accommodate a turbo code. [1] applies to only $(0, k)$ constraint codes and [4] requires either a systematic (d, k) encoder or a $(0, k)$ constraint code; in this paper we consider a general (non-systematic) block (d, k) constraint code.

With the advent of soft-input, soft-output iterative decoders on constrained-input channels, it was shown in [1] that it is possible to recover some of the rate loss accompanying the constraint. In this paper we take a look at whether this recovery is also possible with the configuration in Fig. 1; in connection with this investigation, we ask a question relating to the general class of nonlinear nonsystematic block codes with rate $(n-1)/n$ and $d_{min} = 1$. We illustrate the salient issues with the aid of two motivating examples.

The remainder of the paper is organized as follows: Section 2 addresses the derivation of the log APP ratios for the input bits to a generic block encoder, Section 3 introduces some motivating examples for which soft outputs from some block rate $(n-1)/n$ constraint demodulators are passed to a turbo decoder, Section 4 addresses the maximum likelihood (ML) union bound for nonlinear block codes, Section 5 introduces a simple family of block $(0, 2n-2)$ rate $(n-1)/n$ constraint codes with $d_{min} = 2$, and finally, Section 6 provides concluding remarks.

2 APP Computation for Generic Block Codes

In this section we suggest a method for the computation of the log APP (LAPP) ratio for the input bits of a generic block code. We assume a general (n, k) block code $\mathcal{C}$ for which there is a one-to-one mapping from each distinct k-tuple over $GF2$ ("data bits") to one of the 2^k code words of length n (also over $GF2$). Let one of the 2^k input blocks be represented by $\boldsymbol{d}^{(i)} = (d_0^{(i)}, d_1^{(i)}, \cdots, d_{k-1}^{(i)})$, $i \in \{0, 1, \cdots, 2^k - 1\}$ and the corresponding code word by $\boldsymbol{c}^{(i)} = (c_0^{(i)}, c_1^{(i)}, \cdots, c_{n-1}^{(i)})$. Assume $\boldsymbol{c}^{(i)}$ is BPSK-modulated and transmitted over a noisy channel resulting in the noisy received vector $\boldsymbol{r} = (r_0, r_1, \cdots, r_{n-1})$ from which we would like to compute the LAPP ratio for an input bit $d_j, j \in \{0, 1, \cdots, k-1\}$ defined by

$$L(d_j) = \log \frac{Pr(d_j = 1|\boldsymbol{r})}{Pr(d_j = 0|\boldsymbol{r})} \tag{1}$$

By the usual invocation of Bayes' theorem, we obtain

$$L(d_j) = \log \frac{p(\boldsymbol{r}|d_j = 1) Pr(d_j = 1)}{p(\boldsymbol{r}|d_j = 0) Pr(d_j = 0)}$$

and continuing,

$$L(d_j) = \log \frac{p(r|d_j = 1)}{p(r|d_j = 0)}$$

$$= \log \frac{\sum\limits_{i \in \mathcal{S}_1(j)} p(r|c^{(i)} \text{ transmitted})}{\sum\limits_{i \in \mathcal{S}_0(j)} p(r|c^{(i)} \text{ transmitted})} \tag{2}$$

for the case where the encoder input bits are equiprobable. The disjoint sets $\mathcal{S}_1(j)$and $\mathcal{S}_0(j)$ (for given j) contain the indices i for which $c^{(i)}$ corresponds to a data block $d^{(i)}$ for which $d_j^{(i)} = 1$, and $d_j^{(i)} = 0$, respectively; clearly $\mathcal{S}_0(j) \cup \mathcal{S}_1(j) = \{0, 1, \cdots, 2^k - 1\}$. The last line of (2) encapsulates both the encoder function and the distance properties of the block code. In the most general sense, the encoder function can be expressed as a set of n Boolean functions (one for each code word bit) each of k variables (the data bits). This expression is exact although not practical for code books with large cardinality, since the complexity of the computation escalates exponentially with k (exactly as in the case for an ML decoder). For the nonsystematic codes considered here, we need to pass soft information to a soft-input ECC; we consider the calculation of this next.

When the channel noise is Gaussian and white, (2) takes the form

$$L(d_j) = \log \frac{\sum\limits_{i \in \mathcal{S}_1(j)} \exp\left[-\frac{1}{2\sigma^2} \sum\limits_{m=0}^{n-1} \left(r_m - 2c_m^{(i)} + 1 \right)^2 \right]}{\sum\limits_{i \in \mathcal{S}_0(j)} \exp\left[-\frac{1}{2\sigma^2} \sum\limits_{m=0}^{n-1} \left(r_m - 2c_m^{(i)} + 1 \right)^2 \right]} \tag{3}$$

Results from repeated simulation of this expression on a number of block codes suggests that $L(d_j)$ is also conditionally Gaussian given d_j (even for very short rate 2/3 block codes for which an appeal to the Central Limit Theorem might not be justified). Thus, the normalized $\overline{L(d_j)} = \frac{1}{<L(d_j)>} L(d_j)$ ($< >$ denoting a time average) has identical properties to the soft (noisy) channel output r_m; $\overline{L(d_j)}$ can thus be passed to a soft-input ECC decoder. Alternatively, as an approximation, we could view $L(d_j)$ as the log likelihood ratio (LLR) for the bit d_j which is assumed to have been BPSK-modulated and sent over a fictitious Gaussian channel resulting in $\widehat{r}_j$ so that we could write, using the same idea as in [8] and [9],

$$L(d_j) \approx \frac{2}{\sigma^2} \widehat{r}_j \tag{4}$$

This permits a simple conversion between $L(d_j)$ computed from (2) above and $\widehat{r}_j$ which is actually passed to the turbo decoder. This is actually the method we use in our simulations. Also, since $sgn[L(d_j)]$ from (2) represents the maximum *a posteriori* (MAP) solution for the data bit d_j, it is guaranteed to result in an error rate performance, at the encoder input, at least as good as that of an ML decoder. We have also observed this from many simulations.

If C has $d_{min} \geq 2$, we would expect that, above a certain signal-to-noise ratio (SNR) threshold, the SNR at the output of this soft-output decoder, i.e.,

$1/\mathrm{var}(\overline{L(d_j)}|d_j)$, will be higher than the $\frac{1}{\sigma^2}$ at the channel output (assuming transmission of ±1 over the channel). Most block constraint codes have $d_{min} = 1$ and many are of rate $(n-1)/n$. So, a question that arises naturally is whether such an SNR improvement could be observed for such a block code lacking any distance. We suspect that such a block code together with its mapping of $\boldsymbol{d}^{(i)}$ to the $\boldsymbol{c}^{(i)}$) would have to possess this SNR-improvement property in order for a portion of it's rate loss to be recovered in the configuration of Fig. 1. As an example of how this is not always possible, the block rate $k/(k+1)$ systematic constraint code which enforces a $(0,k)$ constraint by appending a *one* to the end of each k-bit data block could never recover its rate loss of $10\log_{10}((k+1)/k)$ dB in the configuration of Fig. 1 since the error rate at the output of it's very simple soft-input soft-output decoder (which simply removes the indices of $\boldsymbol{r}$ corresponding to the inserted *ones*) is no better (or worse) than that at the channel output.

As pointed out earlier, the need to compute an Euclidean distance between $\boldsymbol{r}$ and each codeword (for a Gaussian channel) makes this computation impractical for codes with large cardinality. It is possible to obtain a suboptimal estimate of the soft $L(d_j)$ by using some method to significantly reduce the size of the set of codewords over which Euclidean distances are computed. Chase [3] demonstrated that there are several algorithms for selecting a good subset of the code book which drastically reduces the required computational burden in exchange for a small performance penalty; the effectiveness of this subset selection procedure is enhanced by a simple, fast method ("go, no-go") of determining if a particular n-tuple is a valid codeword. Pyndiah [7] has used some of these Chase algorithms to obtain soft outputs for turbo codes based on systematic linear block codes. We have implemented several of these types of algorithms with the configuration of Fig. 2 and have observed that the conditional (on knowledge of d_j) pdf of $L(d_j)$ is dependent on the code, the actual Chase algorithm, and the SNR at the channel output. In most cases it is not Gaussian at all. The challenge then, in these cases, is to be able to estimate the conditional pdf since it is required by the BCJR algorithm. A naïve assumption that this conditional pdf is Gaussian might still yield good performance[1]—this point requires further investigation.

3 Two Motivating Examples

Here we provide two examples which provide motivation for further investigation into the possibilities of soft-output APP algorithms for block constraint codes in the configuration of Fig 1.

3.1 A Simple Rate $2/3$ $(0,3)$ $d_{min} = 1$ Code

As a first motivating example, consider the following simple nonlinear rate 2/3

---

[1] The decoding of interleaved serial concatenation in some of the literature appears to make such an assumption since we have never seen this point addressed. The APP decoder for the outer code receives the equivalent of our $\widehat{r}_j$ in (4) from the APP decoder for the inner code (computed with the BCJR algorithm though) and assumes it is conditionally Gaussian when it may not be. In fact when the inner "code" is a precoded PR4 target [9], [10] $\widehat{r}_j$ is not Gaussian at all when d_j is *zero* but [9] and [10] still show very good performance despite this assumption.

(0, 3) block code with $d_{min} = 1$;

$d_0 d_1$	$c_0 c_1 c_2$
0 0	0 1 1
0 1	1 1 1
1 0	0 1 0
1 1	1 0 0

Table 1. Rate 2/3 (0, 3) block code.

From the notation in Section 2, the sets $\mathcal{S}_1(0)$ and $\mathcal{S}_0(0)$ are $\{2,3\}$ and $\{0,1\}$, respectively, while the sets $\mathcal{S}_1(1)$ and $\mathcal{S}_0(1)$ are $\{1,3\}$ and $\{0,2\}$, respectively. For the simple system of Fig. 2, the effective rate loss of the code is only 1.2 dB (as shown in Fig. 3) compared with a nominal $10\log_{10}(3/2) = 1.8$ dB; supplying a modest 0.6 dB of gain from the code/encoder structure. The error rate at the channel output (for c_k) is just that for BPSK signaling adjusted for the code rate loss,

$$P_b^{c_k} = Q\left(\sqrt{2 \cdot \frac{2}{3}\frac{E_b}{N_o}}\right). \tag{5}$$

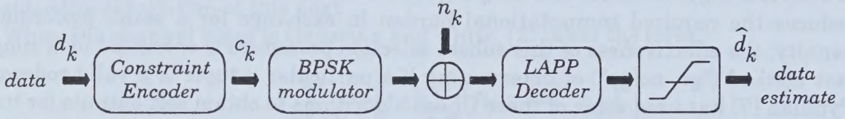


Figure 2. LAPP decoder test configuration.

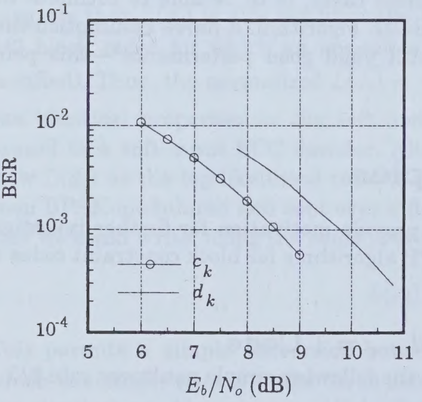


Figure 3. BER comparison for c_k and d_k in system of Fig. 2.

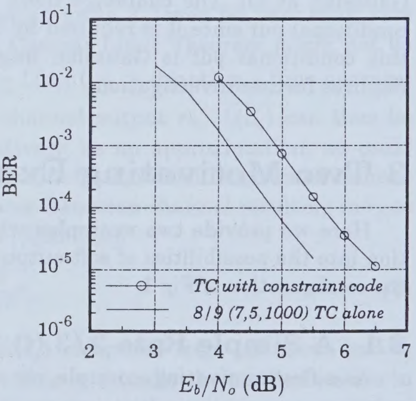


Figure 4. BER performance for configuration of Fig. 1 with 2/3 code.

If we now use this LAPP block as the soft-output constraint decoder in Fig. 1 (with a random interleaver between the ECC and constraint code encoders, and the corresponding deinterleaver between the two decoders — see next paragraph) we obtain the curves in Fig. 4 which reflect an overall net rate loss of only 0.8 dB — instead of the 1.8 dB due to the $d_{min} = 1$ code. So we have recovered 1 dB (just over half the dB amount) of the rate loss of the code despite its low efficiency (70%). This is in contrast to the results from [1] where efficiencies in excess of 90% are required to recover half (the dB amount) of the rate loss. Another interesting point is that, an alternate encoder for the same code exists that recovers only 0.3 dB of the rate loss because it associates minimum distance error events with more data bit errors.

An important point to note concerning the interface between the turbo decoder and the LAPP algorithm is this; within a particular data block, $L(d_0)$ and $L(d_1)$ are not really independent. In the general case there is some correlation between $\{L(d_j)\}_{j=0}^{j=k-1}$ because they are all computed from the same r. The standard implementation of the BCJR algorithm is based on the assumption of independent $L(d_j)$; thus the use of the random interleaver/deinterleaver pair. The matching deinterleaver is therefore placed just after the soft-output APP decoder.

3.2 Two 4/5 $(0, k)$ $d_{min} = 1$ Codes

This example shows how two different codes with the same rate can differ markedly when used in the scheme of Fig. 1. The two codes are listed below together with specific encoders.

d	$C_1(d)$	$C_2(d)$	d	$C_1(d)$	$C_2(d)$
0000	00001	00011	1000	01100	10011
0001	00010	00010	1001	01101	10001
0010	00100	00111	1010	10000	10110
0011	00101	00100	1011	10001	10101
0100	00110	01011	1100	10010	11010
0101	01000	01000	1101	10100	11001
0110	01001	01101	1110	10101	11100
0111	01010	01110	1111	10110	11111

Table 2. Two rate 4/5 $d_{min} = 1$ constraint codes. $C_1(d)$ is the $(0, 8)$ MTR code of Moon & Brickner [6] with a particular encoder; $C_2(d)$ is a $(0, 6)$ code we have designed.

Figs. 5 and 6 show how these codes perform in the configurations of Figs. 1 and 2. We see a larger difference between the performance of these two codes in Fig. 6 than there is in Fig. 5 (2 dB vs. 1.3 dB). We conjecture that the 2 dB difference that appears in Fig. 6 is due to the fact that, in Fig. 5, $C_2(d)$ displays coding gain whereas $C_1(d)$ reflects a "coding loss"—for the system of Fig. 1. So in Fig. 6 $C_1(d)$ suffers an effective rate loss of almost 2 dB whereas $C_2(d)$ *completely* recovers its rate loss. This is an improvement over the case for the 2/3 code in the previous example.

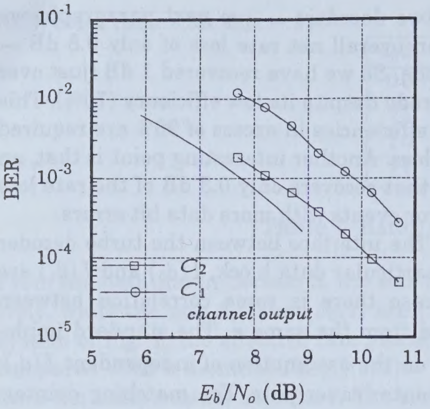


Figure 5. BER comparison (for d_k) between $C_1(d)$ and $C_2(d)$ for system in Fig. 2.

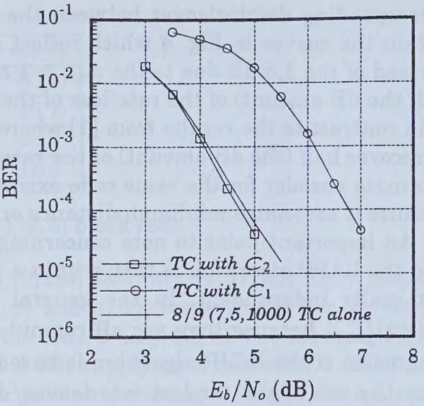


Figure 6. BER comparison for system in Fig. 1.

Both of these examples are interesting in that they show that certain nonlinear block (d,k) codes without distance can be used in such a way that significant amounts of their rate loss can be recovered—at the cost of some complexity increase. Obviously, if the codes possess $d_{min} \geq 2$, we would expect to see the same (partial at least) rate loss recovery effect shown by these two examples. We believe that the behavior exemplified in the two examples is a consequence of both the code and the encoder. In the next section we offer an explanation of this behavior using a simple union bound for nonlinear block codes.

4 An ML Union Bound for Nonlinear Block Codes

If we assume an ML decoder, we can use a union bound to provide some insight into the behavior of nonlinear block codes—those with and without distance. The standard union bound for an (n,k) block code using an ML decoder takes the form

$$P_b \leq \sum_{d=d_{min}}^{n} \beta_d \cdot Q\left(\sqrt{2\frac{k}{n}d\frac{E_b}{N_o}}\right) \tag{6}$$

where P_b is the information bit error rate. We are interested in the distribution of the distance between all pairs of code words and precisely how the encoder maps these code word error events to data block error events. This function is described by β_d. By conditioning the data bit error rate on the transmission of any one of the 2^k code words, we obtain

$$\beta_d = \frac{2\zeta^{(d)}}{k \cdot 2^k} \tag{7}$$

with $\zeta^{(d)}$ denoting the total number of data bit errors associated with all pairs of code words that are Hamming distance d apart, i.e.,

$$\zeta^{(d)} = \sum_{(i,j):\, d_H(c_i,c_j)=d} d_H(d_i, d_j) \tag{8}$$

For codes without any distance, $\beta_1$is of paramount importance, for it 'makes or breaks' the code. It is required to be less than unity in order to avoid losing *more* than the rate loss when the code is used in the system of Fig. 1. The first suggestion that focussing on error coefficients could be very profitable was made very convincingly with the introduction of turbo codes [2]; we make the same observation here. The code used in the first example has $\beta_1 = 1/2$, $\beta_2 = 1$, $\beta_3 = 1/2$. And in the second example, $C_1(d)$ has $\beta_1 = 55/32$ and $\beta_2 = 86/32$ whereas $C_2(d)$ has $\beta_1 = 7/32$ and $\beta_2 = 111/32$. Since these coefficients are for an ML union bound, the error rate for a MAP decoder is almost guaranteed to be better and it is in light of this fact that we can make predictions about the performance of a MAP decoder.

Of course, we would not have to focus as much on error coefficients if we had a constraint code with some distance; in the next section we introduce a family of simple $(0, k)$ block codes with $d_{min} = 2$.

5 A Family of Rate $(n-1)/n, (0, 2n-2)$ Codes with $d_{min} = 2$.

A rate $(n-1)/n$ code uses exactly half of all binary n-tuples, so there are $\binom{n}{n/2}$ such codes. We show a simple method of partitioning the complete set of n-tuples into two equal-size sets. Each of these sets forms a $d_{min} = 2$ code but one contains the all-*zeros* vector; the other constitutes a $(0, 2n-2)$ code. We will designate a code by a matrix whose rows are the code words.

The first observation is that the full set of n-tuples (a matrix) can be assembled from the full set of $(n-1)$-tuples by simply stacking and augmenting smaller matrices as shown below (with the correct size for the column vectors $[\,0\,]$ and $[\,1\,]$) ;

$$\begin{bmatrix} n\text{--}tuples \end{bmatrix} = \begin{matrix} [\,0\,][\,(n-1)\text{--}tuples\,] \\ [\,1\,][\,(n-1)\text{--}tuples\,] \end{matrix} \tag{9}$$

If we pick, from the full matrix of $(n-1)$-tuples, a pair of $(n-1)$-tuples $(\boldsymbol{m}^{(i)}, \boldsymbol{m}^{(j)})$ for which $d_H(\boldsymbol{m}^{(i)}, \boldsymbol{m}^{(j)}) = 1$ and then translate them into a pair of n-tuples by prefixing $\boldsymbol{m}^{(i)}$ with a *zero* and $\boldsymbol{m}^{(j)}$ with a *one*, we now have a pair of n-tuples $(\boldsymbol{n}^{(i)}, \boldsymbol{n}^{(j)})$ with $d_H(\boldsymbol{n}^{(i)}, \boldsymbol{n}^{(j)}) = 2$; this is the general idea behind the construction. The required code is recursively built by beginning with the biphase code (at $n = 2$ this is the smallest code in the family);

$$C_2 = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \text{and } C_2^{*} = \begin{bmatrix} 0 & 0 \\ 1 & 1 \end{bmatrix}$$

where the *conjugate* of C_n, which we designate C_n^{*}, will be described shortly.

Clearly, both codes have $d_{min} = 2$ and any pair of code words $(c_2^{(i)}, c_2^{(j)})$ such that $c_2^{(i)} \in C_2$ and $c_2^{(j)} \in C_2^*$ obviously have $d_H(c_2^{(i)}, c_2^{(j)}) \geq 1$ since $C_2 \cap C_2^* = \emptyset$. So if we form

$$C_3 = \begin{matrix} [0][C_2] \\ [1][C_2^*] \end{matrix} \tag{10}$$

we will ensure that any code word pair $(c_3^{(i)}, c_3^{(j)})$ has $d_H(c_3^{(i)}, c_3^{(j)}) \geq 2$. Also observe that the run length $k = 4$. The construction can be continued recursively by forming

$$c_n^{*(i)} = c_n^{(i)} \oplus c_n^{(j)}, \forall i \in \{0, 1, \cdots, 2^{n-1} - 1\}, \text{for } any\ j \in \{0, 1, \cdots, 2^{n-1} - 1\}$$

that is, the conjugate is formed by adding *any* code word in C_n to all the code words in C_n. The unique set of rows in the conjugate does not depend on which $c_n^{(j)}$ is used in the addition above—using different $c_n^{(j)}$ results in a conjugate whose rows are simply a permutation of those of the 'mother' conjugate. The mother conjugate always results when the *first* row $(c_n^{(0)})$ is used in the addition[2]. So in order to maintain the symmetry we always construct the mother conjugate using $j = 0$ and recursively apply the general form of (10), i.e.,

$$C_n = \begin{matrix} [0][C_{n-1}] \\ [1][C_{n-1}^*] \end{matrix} \tag{11}$$

This completes the construction. Since C_n contains exactly half of all the n-tuples it constitutes a rate $(n-1)/n$ code. One consequence of the construction is that all of the code words in this family have odd weight. This property can be of value in detecting error patterns of odd weight but perhaps more importantly, it can be used to very quickly screen potential code words in an algorithm that computes Euclidean distances to only a few of the code words in the code (see discussion at the end of Sect. 2). Because of the distance of this family of codes, they are guaranteed to outperform the channel error rate in the configuration of Fig. 2.

6 Concluding Remarks

We have tried to show that when true LAPP ratios are computed for some block codes without distance (e.g., constraint codes) and then those values subsequently passed to a turbo decoder, it is possible to observe situations in which the rate loss can be completely recovered. It is also possible to observe situations in which the effective rate loss is significantly worse than that due to the block code. We have tried to explain this behavior with the aid of an ML union bound. The computational complexity involved in obtaining these LAPP ratios is equivalent to that for an ML decoder but we believe this complexity can be significantly reduced by use of variants of the Chase algorithms [3] to approximate an ML solution. If the performance penalty from these approximations is small, this system configuration will offer a very attractive tradeoff between moderately increased complexity and substantial rate loss recovery. The configuration can also make very effective use of a small-distance (e.g., $d_{min} = 2$) block constraint code to provide net coding gain. An important question in connection with this configuration is this: what can we expect from $d_{min} = 1$ block codes as n gets large? Certainly a comparison between the rates 2/3 and 4/5 codes discussed here suggest a favorable trend.

---

[2] The rows of C_n^* form a linear $d_{min} = 2$ code therefore this code is not $(0, k)$-constrained.

References

[1] K. D. Anim-Appiah and S. W. McLaughlin, "Constrained–Input Turbo Codes For $(0, k)$ RLL Channels," *in Proc., 1999 Conference on Information Sciences and Systems*, pp. 420–425.

[2] C. Berroux, A. Glavieux, and P. Thitimajshima, "Near Shannon Limit Error-correcting Coding and Decoding: Turbo-codes," *in Proc. 1993 Int. Conf. Commun.*, pp. 1064–1070.

[3] D. Chase, "A Class of Algorithms for Decoding Block Codes with Channel Measurement Information," *IEEE Trans. Inform. Theory*, Vol. IT-18, No. 1, pp. 170–182, Jan. 1972.

[4] J. L. Fan and J. L. Cioffi, "Constrained Coding Techniques for Soft-input Decoders," *in Proc., 1999 Int. Conf. on Commun.*

[5] B. H. Marcus, P. H. Siegel, and J. K. Wolf, "Finite-State Modulation Codes for Data Storage," *IEEE J. Select Areas Commun.*, Vol. 10, pp. 5–37, Jan. 1992.

[6] J. Moon and B. Brickner, "Maximum Transition Run Codes for Data Storage Systems," *IEEE Transactions on Magnetics*, Vol. 32, No. 5, pp. 3992–3994, Sept. 1996.

[7] R. M. Pyndiah, "Near-Optimum Decoding of Product Codes: Block Turbo Codes," *IEEE Trans. Commun.*, Vol. 46, No. 6, pp. 1003–1010, Aug. 1998.

[8] P. Robertson, "Illuminating the Structure of Code and Decoder of Parallel Concatenated Recursive Systematic (turbo) Codes," *in Proc., GlobeCom '94*, pp. 1298–1303, 1994.

[9] W. E. Ryan, L. L. McPheters, and S. W. McLaughlin, "Combined Turbo Coding and Turbo Equalization for PR4-Equalized Lorentzian Channels," *in Proc., 1998 Conf. on Information Sciences and Systems,* pp. 489–494.

[10] T. Souvignier, A. Friedmann, M. Öberg, P. Siegel, R. E. Swanson, and J. K. Wolf, "Turbo Decoding for PR4: Parallel Versus Serial Concatenation," *in Proc., 1999 Int. Conf. on Commun.*

Low Density Parity Check Codes for Magnetic Recording*

John L. Fan
Stanford University

Arnon Friedmann
Quantum Corporation

Erozan Kurtas
Quantum Corporation

Steven McLaughlin
Georgia Institute of Technology

September 24, 1999

Abstract

Low density parity check (LDPC) codes are error-correcting codes defined by a sparse parity check matrix. When decoded by the message-passing algorithm, these codes can achieve performance comparable to turbo codes. High-rate LDPC codes are considered for an idealized magnetic recording channel consisting of a perfectly equalized EPR4 channel with additive white Gaussian noise (AWGN). The channel is decoded using the BCJR algorithm, which generates probabilistic information for the LDPC decoder. Soft information is then iterated between the channel decoder and the LDPC decoder for improved performance, in analogy to turbo equalization. Results for coding rates 8/9 and 16/17 and block lengths corresponding to a sector of 512 user bytes show that LDPC codes perform close to turbo codes, while requiring less complexity. In addition, since the problem of thermal asperities can be roughly modelled as a burst of erasures, the performance of turbo and LDPC codes against bursts of erasures is considered.

1. Introduction

Turbo codes [2] and low-density parity check codes (LDPC) [5] are error-correcting codes with iterative decoding algorithms that involve the passing of probabilistic information. With these codes, it becomes possible to transmit data error-free at rates extremely close to the full capacity of the channel, where the decoding complexity is substantial but not infeasible. There have been numerous papers on the application of turbo codes to magnetic recording, including [8, 9], which show that for high rates, turbo codes can improve performance by 3 dB to 6 dB on magnetic recording channels, and these gains are roughly the same regardless of the code design (serial or parallel

---

*This work was partially sponsored by NSIC (National Storage Industry Consortium).

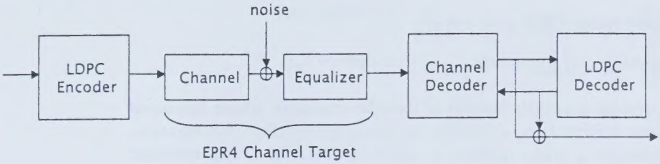


Figure 1.1: Channel block diagram

concatenation) and the channel model (white or non-white noise, ideal or nonideal channel models).

Low-density parity check codes are a more general class of block error-correcting codes, which have been shown to achieve performance close to the theoretical limits, and in particular, they have been shown to perform better than turbo codes as the block length of the code increases. In this paper, we apply a basic LDPC code to magnetic recording, and compare the relative performance of the codes for a magnetic recording situation, where the block length is on the order of a single sector (512 bytes), and a channel decoder is used for the channel intersymbol interference.

We model the channel impulse response as a perfectly equalized partial response polynomial EPR4, where the channel output is given by $y(D) = h(D)\,x(D) + n(D)$ and the channel is $h(D) = 1 + D - D^2 - D^3$. This model corresponds to partial response maximum likelihood (PRML), a common technique in which the channel is equalized to a particular polynomial target. In a realistic system, the noise after equalization will most likely be colored due to the mismatch between the channel response and the partial response polynomial, but in this model, the noise $n(D)$ is assumed to be additive white Gaussian noise. (Refer to [4] for an application of LDPC codes to a more realistic channel modelled by an equalized Lorentzian pulse.)

A block diagram of the system is pictured in Figure 1.1. To decode the channel in a manner compatible with soft iterative decoding, it is necessary to apply an algorithm such as the BCJR algorithm [1] (or forward-backwards algorithm) to decode the intersymbol interference. In analogy to turbo equalization, in which the turbo decoding alternates with the decoding of the channel, soft information is iterated between the LDPC decoder and the channel decoder, which results in a joint decoding of channel and code, but with less much complexity than a maximum-likelihood decoder.

It is useful to review some terminology to describe this decoder. If $p = P(x = 1)$ is the probability that a bit x is equal to 1, then with respect to an observation y, the **prior** probability $p = P(x = 1)$ refers to what was known before the observation, while the **posterior** probability $P(x = 1 \mid y)$is the conditional probability given the observation y. Finally, the **extrinsic** information is what has been learned from the observation y, and is the difference of the posterior and prior information, when expressed as log-likelihood ratios $LLR(x) = \log \frac{p}{1-p}$:

$$\text{LLR}^{\text{prior}}(x) + \text{LLR}^{\text{extrinsic}}(x) = \text{LLR}^{\text{posterior}}(x)$$

The iterative decoding system generally consist of decoder modules, where the input to each module is the prior information, while the output is the extrinsic information, and the prior information for a given module is equal to the sum of the extrinsic information from the other modules.

In the block diagram in Figure 1.1, the channel decoder begins by taking the received channel samples (and no prior information on the bits) and producing output probabilities which are then passed as prior information into the decoder for the LDPC code. The LDPC decoder operates on this data using the message-passing algorithm, as described in the next section, generating extrinsic information which can then be passed back to the BCJR channel decoder as prior information. As in the case of turbo codes, iterating back and forth between the two decoders gives better performance than only applying the channel decoder once.

2. Low-Density Parity Check codes

A binary low-density parity check (LDPC) code is a binary linear code defined by a sparse parity check matrix. Suppose that the parity check matrix A has N columns and M rows. Then the codeword consists of N bits which satisfy M checks, where the location of a 1 in the parity check matrix indicates which that a bit is involved in a parity check. The total length of the codeword is N bits, the number of message bits is $K = N - M$, and the rate of the code is $\frac{K}{N}$, assuming that the matrix is full rank. As shown in Figure 2.1, it is possible to associate a bipartite graph to this parity check matrix, consisting of N bit nodes (indicated by the white circles) and M check nodes (indicated by the black squares), with an edge between the bit and check nodes if there is a 1 in the parity check matrix. It is known how to design irregular parity check matrices (ref. [7]) to achieve performance extremely close to the Shannon limit of the channel. In this paper, we consider regular parity check matrices, where each column has a fixed number of 1's and each row has a fixed number of 1's, since this serves as a common benchmark of performance.

2.1. Message passing algorithm

The message-passing algorithm, also known as the sum-product algorithm or belief propagation, is a soft iterative decoding algorithm which passes messages along the edges of this bipartite graph. Following [6], the message-passing algorithm takes the following form: Let (r_{ij}^0, r_{ij}^1) be a pair of probabilities which is passed from the check node f_j to the bit node x_i. Similarly, let (q_{ij}^0, q_{ij}^1) be the probabilities sent from a bit node x_i to a check node f_j. For each bit, let p_i^1 denote the prior probability that $x_i = 1$ from the channel decoder, and $p_i^0 = 1 - p_i^1$. Then the updates take the form

$$q_{ij}^0 = \alpha_{ij} p_i^0 \prod_{j' \in \text{Col}[i] \setminus \{j\}} r_{ij'}^0$$

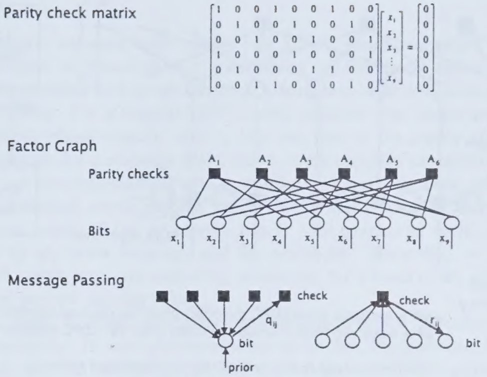

Parity check matrix
Factor Graph
Parity checks
A_1 A_2 A_3 A_4 A_5 A_6
Bits
x_1 x_2 x_3 x_4 x_5 x_6 x_7 x_8 x_9
Message Passing
check
q_{ij}
bit
prior
check
r_{ij}
bit

Figure 2.1: Parity check matrix and bipartite graph for LDPC codes

$$q_{ij}^1 = \alpha_{ij} p_i^1 \prod_{j' \in \mathrm{Col}[i] \backslash \{j\}} r_{ij'}^1$$

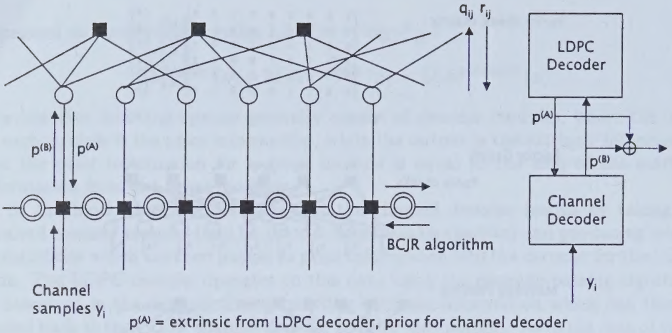
The set $\mathrm{Col}[i] \backslash \{j\}$ indicates all the row locations in the i-th column of the matrix which contain a 1, excluding the j-th row. In other words, if the bit node x_i participates in n checks, then the message to the check node f_j depends only on the incoming messages from the $n-1$ other check nodes along with the prior information. On the other hand, the update equation for the messages from checks to bits is given by $r_{ij}^0 - r_{ij}^1 = \prod_{i' \in \mathrm{Row}[j] \backslash \{i\}} \left(q_{i'j}^0 - q_{i'j}^1\right)$ where $\mathrm{Row}[j] \backslash \{i\}$ indicates the column locations in the j-th row of the matrix which contain a 1, excluding the column i. Since $r_{ij}^0 + r_{ij}^1 = 1$, this gives the pair of equations:

$$r_{ij}^0 = \frac{1}{2}\left(1 + \prod_{i' \in \mathrm{Row}[j] \backslash \{i\}} \left(q_{i'j}^0 - q_{i'j}^1\right)\right)$$

$$r_{ij}^1 = \frac{1}{2}\left(1 - \prod_{i' \in \mathrm{Row}[j] \backslash \{i\}} \left(q_{i'j}^0 - q_{i'j}^1\right)\right)$$

Finally, the extrinsic output of the LDPC decoder is given by

$$e_i^0 = \alpha_i \prod_{j' \in \mathrm{Col}[i]} r_{ij'}^0$$



$p^{(A)}$ = extrinsic from LDPC decoder, prior for channel decoder
$p^{(B)}$ = extrinsic from channel decoder, prior for LDPC decoder

The final output is the sum of the log-likelihood ratios:
$\text{LLR}(x_i) = \text{LLR}(p_i^{(A)}) + \text{LLR}(p_i^{(B)})$

Figure 2.2: Message passing between the channel decoder (BCJR algorithm) and the LDPC decoder

$$e_i^1 = \alpha_i \prod_{j' \in \text{Col}[i]} r_{ij'}^1$$

where the prior probabilities have been omitted. The constant α_i ensures that $e_i^0 + e_i^1 = 1$. This extrinsic probability is passed to the channel decoder as prior information for the BCJR algorithm.

In Figure 2.2, the details of message passing between the channel decoder and the LDPC decoder are shown. The extrinsic output of the BCJR algorithm is denoted by $p_i^{(B)}$, and this becomes the prior probability p_i^1 for the LDPC decoder. On the other hand, the output extrinsic probability e_i^1 from the LDPC decoder is denoted by $p_i^{(A)}$. The iteration scheme goes as follows: the channel decoder processes the received signal, and passes $p^{(B)}$ to the LDPC decoder, which generates its own extrinsic output $p^{(A)}$. This is passed back as prior information for the channel decoder, which then decodes again. The updated output of the BCJR algorithm is the new $p^{(B)}$, which is taken as the updated prior information for the LDPC decoder. It should be noted that at this stage, to calculate the messages $\{q_{ij}\}$, it is necessary to use the messages r_{ij} from the previous iteration. This differs from turbo codes, where each iteration of the BCJR algorithm for the constituent codes which operates only on the prior information into the decoder, and does not preserve computated values from the previous iteration.

2.2. Complexity

Making a comparison between turbo codes and LDPC codes, we count the number of operations involved in decoding. By considering log-likelihood ratios rather than probabilities, it is possible to convert multiplications into additions using the appropriate transformations. For a regular LDPC code, suppose that there are t_c bits per column in the parity check matrix, and t_r bits per row in the parity check matrix. Each of the operations for a message from bit to check requires the addition of $t_c + 1$ numbers, which can be implemented with t_c additions of real numbers, and then subtracting off the incoming message from check to bit, for a total of $2Nt_c$ additions. On the other hand, each of the M checks has t_r bits connected to it, so each can be implemented by $4t_r$ table look-up, and $2t_r$ additions. Since $Mt_r = Nt_c$, this is equivalent to $4Nt_c$ table look-ups and $2Nt_c$ additions, for a total of $4t_c$ additions and $4t_c$ table look-ups per bit per iteration.

On the other hand, a rough analysis of one implementation of turbo codes gives the following complexity. In one implementation of the BJCR algorithm, for a $S = 2^{\nu}$ state trellis, there are $12S$ additions, $4S$ table look-ups and $4S$ compare operations per bit per iteration. In the simulations in this paper, we use $t_c = 3$ bits per column, and 20 iterations, so that the total number of operations is 24 per bit per iteration, or 480 per bit in total. The turbo codes are run for 10 iterations, and are based on $S = 16$ state convolutional codes, so that the number of operations for a serial concatenated turbo code is $20S = 320$ operations per bit per iteration, and 3200 operations per bit in total. The number of operations doubles to 6400 for the case of a parallel concatenated turbo code. It should be noted that these computations only include the complexity of the LDPC and turbo decoders, and not the BCJR decoder for the channel. It is possible to reduce this complexity by decreasing the number of iterations and optimizing the algorithms, but that analysis is beyond the scope of this paper.

3. Simulations

Simulations of LDPC codes and turbo codes were performed for parameters relevant to hard disk drives. In particular, we considered a block of message bits of length $K = 4352$, corresponding to a sector of 512 user bytes that is slightly expanded due to a modulation code. (No modulation code is actually applied here, although in a practical application of turbo and LDPC codes to magnetic recording systems, the modulation constraint must be considered. A general framework for using modulation constraints with soft iterative decoding is presented in [3]) We consider two possible code rates for the LDPC and turbo codes. In the case of an 8/9 rate code, the overall block length is $N = 4896$, while in the case of a 16/17 rate code, the overall block length is $N = 4624$.

In Figure 3.1, we consider the relative performance on a channel without intersymbol interference. Performance is compared for both rates 8/9 and 16/17, where the SNR has been adjusted for rate by multiplying by $\frac{1}{r}$. As a function of the SNR, we see that the performance curves are relatively close, with the LDPC code performs

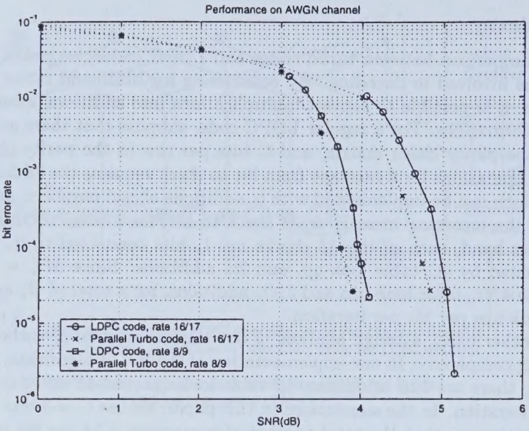


Figure 3.1: Performance on AWGN channel

slightly worse than the turbo code.

In Figure 3.2, the intersymbol interference channel is modelled as an ideal EPR4 channel, and a channel decoder is used to generate soft information. In the first approach, the BCJR channel decoder is applied only once, and in the second approach, soft information is iterated between the channel decoder and the decoder for the LDPC or turbo code. This second approach is denoted "with channel iteration", and is the same as "turbo equalization". A parallel concatenated turbo code consists of two concatenated convolutional codes separating by a random permuter (known as an interleaver), and this can be done with or without channel iteration. A serial concatenated turbo code consists of a single convolutional code, which is concatenated with the channel decoder, so that the decoding involves iteration with the channel. We see that the parallel turbo code with iteration performs slightly worse than the serial turbo code. In addition, the LDPC code with channel iteration performs as well as the serial concatenated turbo code, affirming that the LDPC code are as good as turbo code for magnetic storage systems.

3.1. Thermal asperities

A problem with many magnetic recording read heads is the occurrence of thermal asperities. As a rough approximation, it is possible to model this phenomenon by an extended burst of erased samples. This corresponds to the use of some mechanism that detects that the thermal asperity erases a block of samples to avoid overloading the decoder with errors. We perform simulations in which every sector suffers a burst of

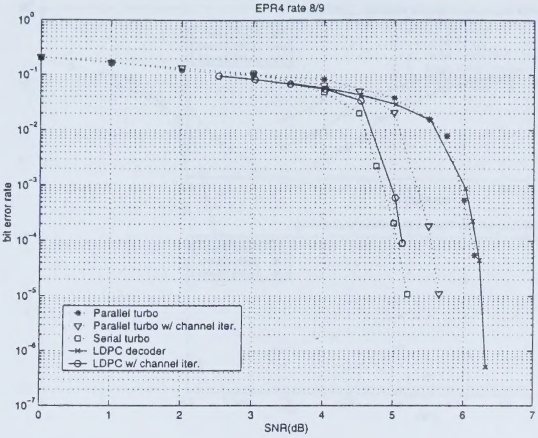


Figure 3.2: Performance of rate 8/9 codes on EPR4 channel

erasures, corresponding to the thermal asperity. First, we consider the performance on an AWGN channel in Figure 3.3. Both LDPC codes and parallel concatenated turbo codes are robust to erasures, showing only a modest coding loss even for a burst of 80 erasures.

On the other hand, in Figure 3.4, we see that for the EPR4 channel, both codes do not perform well when a thermal asperity occurs. The parallel turbo code performs slightly better, however, for lengths of 40 and 80.

Finally, in Figure 3.5, the serially concatenated turbo scheme performs badly in the presence of thermal asperities. This is very likely due to use of a single convolutional code, which has trouble decoding the erased bits. Hence the LDPC code is an attractive coding alternative to serially concatenated turbo codes, especially in the presence of thermal asperities.

References

[1] L. Bahl, J. Cocke, F. Jelinek, and J. Raviv, "Optimal decoding of linear codes for minimizing symbol error rate," *IEEE Trans. Inform. Theory*, Mar. 1974, pp. 284-287.

[2] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon limit error- correcting coding and decoding: Turbo codes," *Proc. 1993 Int. Conf. Commun.*, pp. 1064-1070.

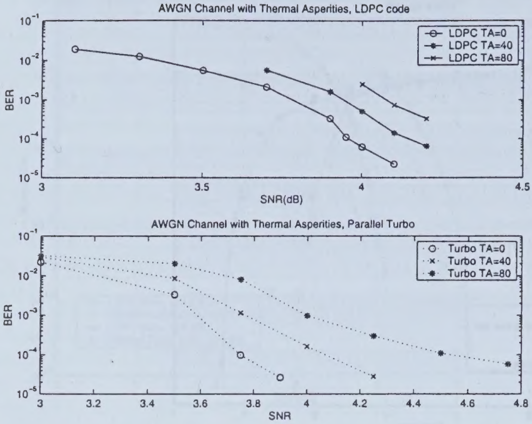


Figure 3.3: AWGN Channel with Thermal Asperities, Rate 8/9

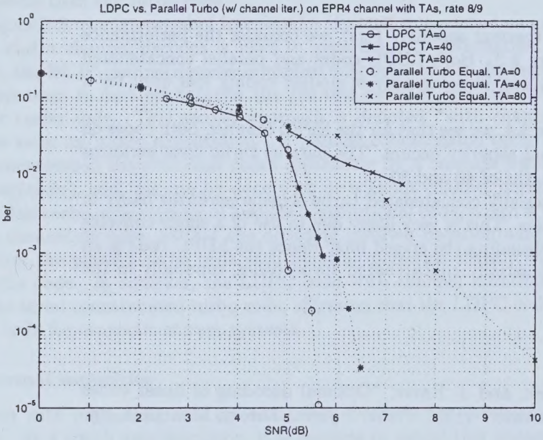


Figure 3.4: EPR4 Channel with Thermal Asperities, LDPC vs. Parallel Turbo with channel iteration

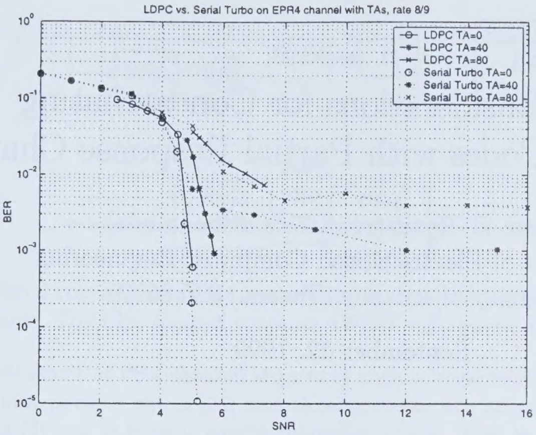


Figure 3.5: EPR4 Channel with Thermal Asperities, LDPC vs. Serial Turbo

[3] J.L. Fan and J.M. Cioffi, "Constrained coding techniques for soft iterative decoding," Globecom '99 (Rio de Janeiro, Brazil).

[4] J.L. Fan, E. Kurtas, A. Friedmann and S. McLaughlin, "Low Density Parity Check Codes for a PR-Equalized Lorentzian Recording Channel," submitted to ICC 2000.

[5] R.G. Gallager, "Low-Density Parity Check Codes," *IRE Trans. Inform. Theory,* January 1962, pp. 21-28.

[6] D.J.C. MacKay, "Good Error-Correcting Codes Based on Very Sparse Matrices," *IEEE Trans. Inform. Theory,* Vol. 45, No. 2, March 1999, pp. 399-431.

[7] T. Richardson, A. Shokrollahi and R. Urbanke, "Design of Provably Good Low-Density Parity Check Codes," to appear in *IEEE Trans. Inform. Theory.*

[8] W. Ryan, L. McPheters and S. McLaughlin, "Combined Turbo Coding and Turbo Equalization for PR4-Equalized Lorentzian Channels," *Proc. of Conf. Inform. Sci. and Sys.*, Princeton, NJ, Mar. 1998.

[9] T. Souvignier, A. Friedmann, M. Oberg, P. Siegel, J. K. Wolf, " Turbo codes for PR4: serial versus parallel concatenation," *Proceedings of the 1999 International Conference on Communications*, Vancouver, BC, Canada, June 1999.

Design Considerations for Concatenating Convolutional Codes with Partial Response Channels*

William E. Ryan (*ryan@ece.arizona.edu*)
Department of Electrical and Computer Engineering
University of Arizona, Tucson, AZ 85721

September 22, 1999

Abstract

We present design principles for serial concatenations of convolutional codes with partial response channels, focusing in particular on the Class IV partial response (PR4) channel. Code rates of the form $K_0/(K_0+1)$, for $K_0 = 4, 8, 16, 32$, and 64 are considered. We show that the $\frac{1}{1\oplus D^2}$ precoder, first proposed by Wolf and Ungerboeck for trellis-coded PR4 channels, produces a relatively high error rate floor. This motivates new interleaver and precoder designs, the latter involving an examination of the *conditional distance enumerator* of each precoded PR4 channel. The new designs substantially reduce the level of the floors and allow the use of a simpler outer convolutional code.

1. Introduction

The recent discovery of concatenated codes with iterative (turbo) decoding [1] has led to a new coding paradigm which has been successfully applied to partial response systems in [2]-[8], where coding gains substantially greater than those of trellis codes have been reported for both parallel and serial concatenated systems. (Related work can also be found in [9].) In the present work, we take a closer look at serial concatenated code (SCC) designs for partial response systems, where the outer code is a recursive systematic convolutional (RSC) code and the inner code is a partial response channel. The decoder is an iterative BCJR decoder [10] with two component decoders sharing *a posteriori* probability (APP) information iteratively. Our focus is on Class IV partial response (PR4) modeled by the polynomial $1 - D^2$, although the approaches proposed here can be extended to other forms of partial response in a straightforward manner. The entire coded PR4 system is depicted in Fig. 1 where we make note of the fact that the RSC code is punctured to achieve code rates of the form $K_0/(K_0+1)$, for $K_0 = 4, 8, 16, 32$, and 64.

The initial code designs for PR4 systems have generally employed the (Tomlinson) precoder whose transfer function is $\frac{1}{1\oplus D^2}$. This particular precoder has the effect of ensuring that binary codes with a large minimum Hamming distance applied to the PR4

---

*This work was supported in part by a National Science Foundation grant and a gift from the National Storage Industry Consortium.

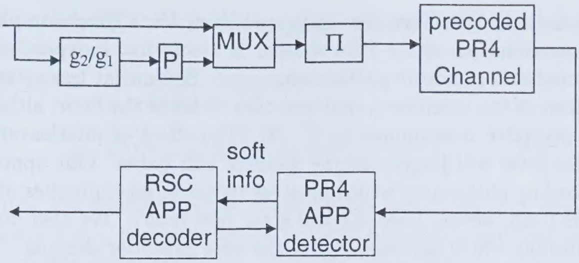


Fig. 1. Serial concatenated PR4 system and its iterative decoder. (The 'P' indicates a puncturer. The PR4 channel contains AWGN with power spectral density $N_0/2$.

channel will result in PR4 channel sequences with a large minimum Euclidean distance. This was first pointed out in [11] and this approach was adopted for turbo-coded PR4 channels in [2] and in several papers thereafter [3]-[8]. This is apparently the best precoder for parallel concatenations, but it can be substantially improved upon for serial concatenations by alternative precoder designs as the SCC's suffer from a relatively high error rate floor.

To demonstrate this point, consider rate $K_0/(K_0+1)$ SCC's with 4096-bit word sizes (hence, interleaver sizes $4096(K_0+1)/K_0$). The RSC code employs the polynomials (g_1, g_2) = (23, 31) and the rate $K_0/(K_0+1)$ is achieved by saving the second bit in every K_0-bit parity block of the RSC encoder output. The performance of this system with the $\frac{1}{1 \oplus D^2}$ precoder is depicted in Fig. 2 where it is seen that an error rate floor starts to show its presence somewhat early in the error rate curve for the code rates considered.

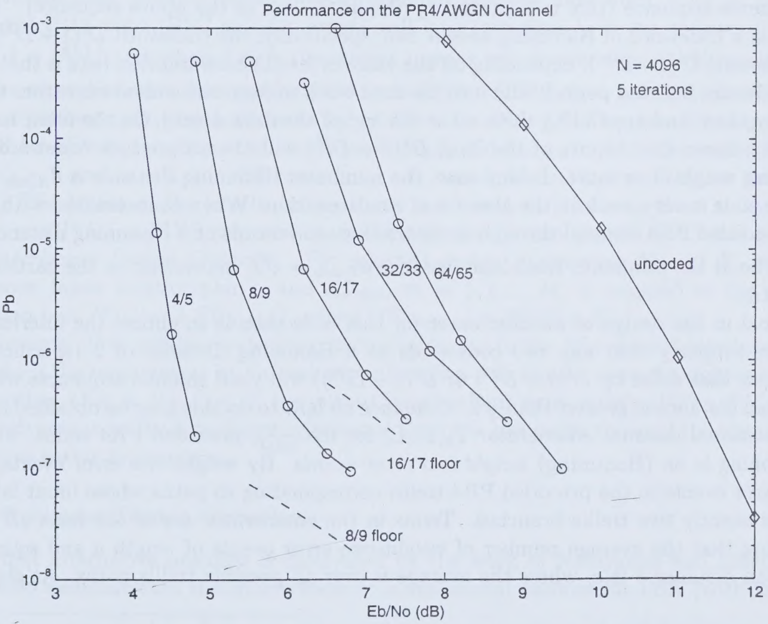


Fig. 2. Performance of SCC's on the $\frac{1}{1 \oplus D^2}$-precoded PR4 channel.

These initial code designs have also indicated that, for a fixed complexity, the performance of a concatenated coded PR4 system is somewhat independent of the code polynomials selected and puncture pattern employed. Somewhat less attention has been given to the abilities of the interleaver and precoder to lower the floor, although the effect of removing the precoder is examined in [7], [8]. The effect of interleaver and precoder design on the floor level will largely be the focus of this paper. Our approach is to embrace the turbo coding philosophy which involves reducing multiplicities of the minimum weight codewords (and, hence, lowering the error rate floor). We also discuss an interleaver design criterion which accommodates the new precoder designs. The combined effect of the new interleaver-precoder combinations are performance curves with vastly reduced error rate floors.

Section 2 establishes interleaver and precoder design criteria which lead to improved concatenated code performance. Section 3 then presents selected performance results, demonstrating substantial gains over the initial code designs.

2. Improved Interleaver and Precoder Designs

2.1. Interleaver Design Criterion

We introduce the interleaver design criterion by way of example. Consider the rate 8/9 SCC of the previous section whose RSC code polynomials are $g_1(D) = 1 + D^3 + D^4$ and $g_2(D) = 1 + D + D^4$. Observe that an RSC encoder input of the form $D^k(1+D^{30})$ will result in the parity output $D^k(1+D^{30})g_2(D)/g_1(D) \leftrightarrow$ ...00$\underline{0}$1101111$\underline{0}$0010011$\underline{0}$1011110$\underline{0}$010 0111$\underline{0}$00... which is periodically "sampled" by the puncturer. No matter what the sampling phase is, there will always exist a shift k such that punctured parity sequence is the all-zeros sequence (this is indicated by the underbars in the above sequence). This results in a codeword of Hamming weight two, specifically, the codeword $D^l(1 + D^{33})$ or the codeword $D^l(1 + D^{34})$, depending on the value of k. (The assumption here is that the parity bits are inserted periodically into the data word to form the codeword, rather than buffering them and appending them all at the end of the data word.) On the other hand, it can be shown that inputs of the form $D^k(1 + D^{15})$ will always produce codewords of Hamming weight 3 or more. In any case, the minimum Hamming distance is $d_{H,\min} = 2$ and the code is very weak in the absence of concatenation. When concatenated with the $\frac{1}{1\oplus D^2}$-precoded PR4 channel through an interleaver, codewords at a Hamming distance of 2 could be at the minimum Euclidean distance $d_{E,\min} = \sqrt{2}$, depending on the particular codewords.

A goal in the design of an interleaver for this code then is to endow the interleaver with the property that any two codewords at a Hamming distance of 2 (specifically, codewords that differ by $D^l(1 + D^{33})$ or $D^l(1 + D^{34})$) will yield channel sequences whose Euclidean distance is greater than $\sqrt{2}$. Guidance on how to do this may be obtained from the *conditional distance enumerator* $T_2(D, L)$ for the $\frac{1}{1\oplus D^2}$-precoded PR4 trellis, where conditioning is on (Hamming) *weight-two error events*. By weight-two error events, we mean error events in the precoded PR4 trellis corresponding to paths whose input labels differ in exactly two trellis branches. Terms in the enumerator are of the form cL^aD^b, indicating that the average number of weight-two error events of length a and squared Euclidean distance b is c, where the average is over all possible trellis paths. As shown in the next subsection, $T_2(D,L)$ for the $\frac{1}{1 \oplus D^2}$-precoded PR4 channel is given by[1]

$$\begin{aligned} T_2(D,L) &= \frac{L^3D^2}{1 - L^2\frac{1}{2}(1+D^4)} \qquad (1) \\ &= L^3D^2 + L^5D^2\frac{1}{2}(1+D^4) + L^7D^2\left[\frac{1}{2}(1+D^4)\right]^2 + \ldots \end{aligned}$$

Thus, for example, the average number of weight-two error events of length 3 and squared Euclidean distance 2 is 1, and the average number of weight-two error events of length 5 and squared Euclidean distance 6 is 1/2. Note that there exist no even-length, weight-two error events (i.e., two trellis paths whose input labels differ in two branches can never remerge in an even number of trellis branches).

This knowledge of the conditional distance enumerator $T_2(D,L)$ indicates that any interleaver which converts inputs of the forms $D^l(1+D^{33})$ or $D^l(1+D^{34})$ to even-length outputs of the form $D^m(1+D^{2k+1})$ would perform well for the rate 8/9 SCC code under consideration. However, this interleaver design criterion is too restrictive, making the interleaver design very difficult and perhaps "at risk" for other types of inputs. Instead, we notice that $T_2(D,L)$ indicates for the general case that the average number of $d_{E,\min}$ weight-two error events of length $2k+3$ is 2^{-k} $(k = 0,1,2,\ldots)$. Hence, the longer the error event, the less likely it is to be a $d_{E,\min}$ event. This leads us to an alternative design criterion which requires that interleaver inputs of the form $D^l(1+D^{33})$ or $D^l(1+D^{34})$ be mapped to outputs of the form $D^m(1+D^n)$, where n is "large." It is easy to see that this type of criterion is useful also for other code rates. We will see that is applicable as well for the alternative precoders to be introduced below.

To state the criterion for SCC's for general code rates of the form $K_0/(K_0+1)$, we first notice that, because parity bits are inserted into the data word, a bit at position k in the data word (RSC encoder input) will be at position l_k in the codeword, where $l_k = k + \left\lfloor \frac{k-1}{K_0} \right\rfloor$ and where $\lfloor x \rfloor$ is the largest integer not greater than x. Note also that we are concerned with data words of the form $D^k(1+D^{15m})$, $m = 1,2,3,\ldots$, for these are the weight-two polynomials divisible by $g_1(D) = 1 + D^3 + D^4$. The data $\mathbf{1}$'s at positions k and $k+15m$ in the data word are at positions l_k and l_{k+15m} in the codeword. (Observe for $K_0 = 8$ and $m = 2$, the difference $l_{k+30} - l_k$ equals 33 or 34, depending on the value of k.) The action of the interleaver on these two locations is to map them to locations $P(l_k)$ and $P(l_{k+15m})$. We may now state the criterion for the general SCC code rate.

Interleaver Design Criterion. For selected design parameters M and S, each interleaver input location-pair l_k and l_{k+15m}, $m = 1,2,\ldots,M$, is mapped to the output location-pair $P(l_k)$ and $P(l_{k+15m})$, where $|P(l_{k+15m}) - P(l_k)| > S$.

Remark. This interleaver design criterion is similar to the "S-random" interleaver [12] for which the requirement is: for each input location-pair l and l' in an S-neighborhood of each other, that is, $|l - l'| < S$, the output location-pair must satisfy $|P(l) - P(l')| > S$. One advantage to the new design criterion is that it is less restrictive and the interleavers are thus easier to generate.

2.2. Precoder Design Criterion

Our first alternative precoder is motivated by the work of Narayanan and others [13]-[15] who demonstrated the effectiveness of a differential encoder for PSK (with transfer function $\frac{1}{1\oplus D}$) acting as a rate 1 recursive inner code in a serial concatenation (almost any outer code will do). Its effectiveness stems from the fact that odd-weight inputs into the differential encoder produce sequences that never remerge with the all-zeros trellis path for the concatenated code, and that, except for trivial cases, sparse even-weight inputs increase in weight at the output. For example, ...0010000100... becomes ...0011111000... and ...001010010000100... becomes ...00110011111000... . Thus, whenever the inner code produces a low weight word, there is a good chance that after random interleaving the differential encoder will produce a large weight codeword. If an interleaver with the above design criterion is employed, the weight-two RSC codewords corresponding to the design parameter M are guaranteed to produce large weight words at the differential encoder output. Thus, we can expect that the multiplicities of low weight codewords in the combined RSC-code/interleaver/differential-encoder to be very low.

---

[1] The conditional weight enumerators here and below assume iid inputs, justified by the presence of the interleaver and the use of very high rate codes.

We can combine the effect of the $\frac{1}{1\oplus D}$ differential encoder with that of the $\frac{1}{1\oplus D^2}$ precoder to great benefit. Since the former has the effect of enhancing the Hamming distance spectrum and the latter has the effect of converting large Hamming distances to large Euclidean distances, the combined effect is a set of channel sequences with an enhanced Euclidean distance spectrum. Before we present simulation results for this code to demonstrate its effectiveness, we first present a unified approach to studying precoders for the PR4 channel, with particular emphasis on weight-two precoder inputs since we are interested in highly punctured RSC codes.

We start with the $\frac{1}{1\oplus D^2}$-precoded PR4 channel whose diagram is given in Fig. 3(a). In Fig. 3(b), we have redrawn it as a rate 1/2 convolutional code followed by a mapper, i.e., as a trellis code. Thus, we may use the distance spectrum techniques used for trellis codes to derive the distance spectrum of this precoded PR4 channel. In particular, the mapper is an isometry that maps the elements of the group { (0,0), (1,0) } and the elements of its coset { (1,1), (0,1) } to signal-point subsets that are 180^o rotations of each other: { 0, 1} and { 0, -1 }, respectively. Hence,we may use the method introduced by Zehavi and Wolf [16] to obtain the Euclidean distance spectrum.

The technique involves first producing the *error state diagram* derived from the usual state transition diagram as depicted in Fig. 3(c). In the diagram, the second branch label indicates the bit error corresponding to the error state diagram transition and the first labels are defined as follows:

$$\begin{aligned}
W(11) &= \frac{1}{2}D^{\|m(00)-m(00\oplus 11)\|}+\frac{1}{2}D^{\|m(10)-m(10\oplus 11)\|}\\
&= \frac{1}{2}+\frac{1}{2}D^4\\
W(10) &= \frac{1}{2}D^{\|m(00)-m(00\oplus 10)\|}+\frac{1}{2}D^{\|m(10)-m(10\oplus 10)\|}\\
&= D\\
W(01) &= \frac{1}{2}D^{\|m(00)-m(00\oplus 01)\|}+\frac{1}{2}D^{\|m(10)-m(10\oplus 01)\|}\\
&= D\\
W(00) &= \frac{1}{2}D^{\|m(00)-m(00\oplus 00)\|}+\frac{1}{2}D^{\|m(10)-m(10\oplus 00)\|}\\
&= 1
\end{aligned}$$

where $m(\cdot)$ is the mapping function. Because we are presently interested in weight-two error events, we prune the error state diagram of Fig. 3(c) to obtain that of Fig. 3(d) which represents all possible weight-two error events. We have also included the factor

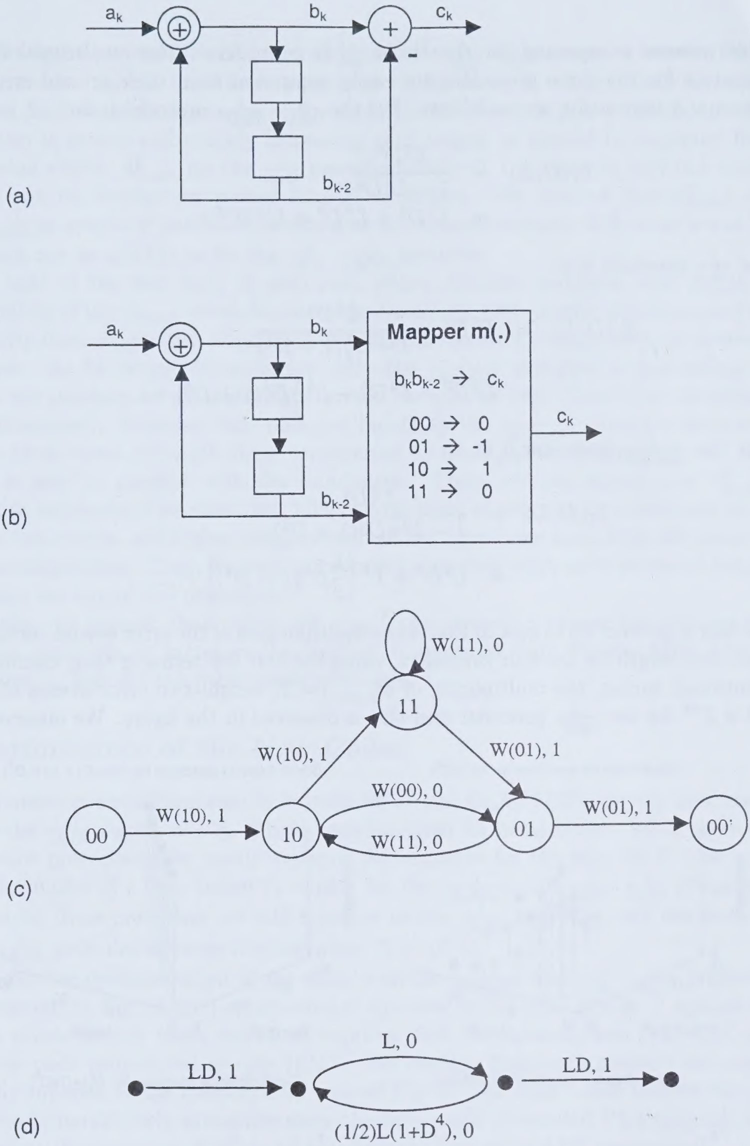

a_k
b_k
c_k
+
-
b_k-2
(a)
a_k
b_k
Mapper m(.)
b_k b_k-2 → c_k
00 → 0
01 → -1
10 → 1
11 → 0
c_k
b_k-2
(b)
W(11), 0
11
W(10), 1
W(01), 1
W(00), 0
W(10), 1
W(01), 1
00
10
01
00'
W(11), 0
(c)
L, 0
LD, 1
LD, 1
(1/2)L(1+D^4), 0
(d)

Fig. 3. Development of the conditional distance enumerator for the $\frac{1}{1\oplus D^2}$-precoded PR4 channel.

L in the first branch labels to provide bookkeeping on the length of error events. The transfer function for this pruned error state diagram is easily computed and is given in equation (1).

This process is repeated for the three other precoders. The conditional distance enumerators for the three precoders are easily computed from their pruned error state diagrams and the results are as follows. For the $\frac{1}{1\oplus D}\cdot\frac{1}{1\oplus D^2}$ precoder it is

$$\begin{aligned} T_2(D,L) &= \frac{L^5D^4}{1-L^4D^4} \\ &= L^5D^4 + L^9D^8 + L^{13}D^{12} + \ldots \quad ; \end{aligned} \tag{2}$$

for the $\frac{1}{1\oplus D}$ precoder it is

$$\begin{aligned} T_2(D,L) &= L^3D^2 + \frac{L^4D^4}{1-L\frac{1}{2}(1+D^4)} \\ &= L^3D^2 + L^4D^4 + L^5\frac{1}{2}D^4(1+D^4) + \ldots \quad ; \end{aligned} \tag{3}$$

and for the $\frac{1}{1\oplus D\oplus D^2}$ precoder it is

$$\begin{aligned} T_2(D,L) &= \frac{L^4D^4}{1-L^3\frac{1}{2}D^2(1+D^4)} \\ &= L^4D^4 + L^7\left[\frac{1}{2}D^6(1+D^4)\right] + \ldots \quad . \end{aligned} \tag{4}$$

We find it instructive to plot in Fig. 4 the multiplicities of the error events versus event distance and length for the four precoders, using the first few terms in their enumerators. As mentioned earlier, the multiplicity of $d^2_{E,\min}$ $(=2)$ weight-two error events of length $2k+3$ is 2^{-k} for the $\frac{1}{1\oplus D^2}$ precoder and this is observed in the figure. We observe also

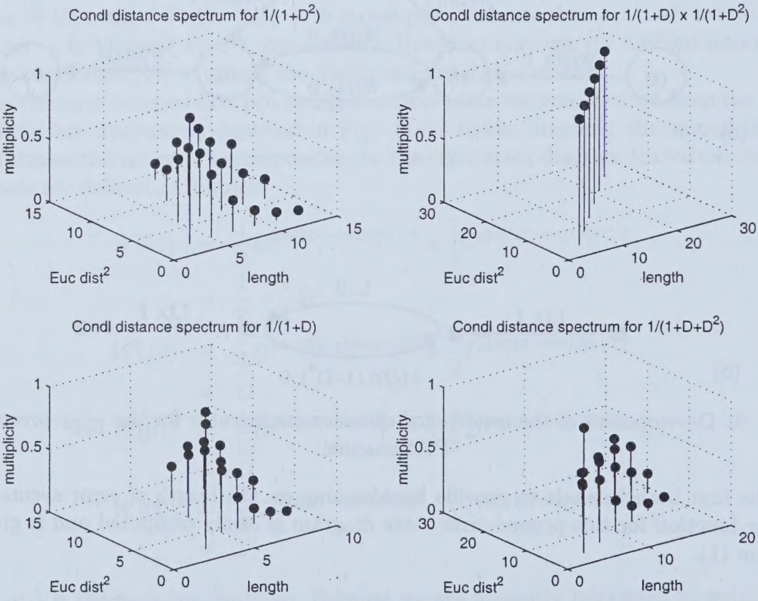


Fig. 4. Depiction of the conditional distance enumerators for the four precoders.

from the figure (or equation (2)) that $d^2_{E,\min}$ for the $\frac{1}{1\oplus D}\cdot\frac{1}{1\oplus D^2}$ precoder is twice that of the $\frac{1}{1\oplus D^2}$ precoder, that its error event lengths are of the form $4k+1$, and their corresponding squared Euclidean distances are of the form $4k$, for $k = 1, 2, 3, \ldots$. Hence, the distance spectrum is sparse and quickly increasing with length as should be expected from the discussion above. $d^2_{E,\min}$ for the $\frac{1}{1\oplus D}$ precoder is only 2, but there is only one such event and it can be avoided by proper interleaver design. We observe that $d^2_{E,\min} = 4$ for the $\frac{1}{1\oplus D\oplus D^2}$ precoder and that its distance spectrum increases with error event length, although not as quickly as for the $\frac{1}{1\oplus D}\cdot\frac{1}{1\oplus D^2}$ precoder.

In light of the fact that, in each case, either distance increases with length or the multiplicity of the $d_{E,\min}$ event decreases exponentially with length, we can expect relative immunity from weight-two error events for interleavers that comply with the above design criterion. As for weight-three events, only the $\frac{1}{1\oplus D\oplus D^2}$ precoder is susceptible, as the others are immune to all odd-weight events (as can be seen from their complete error state diagrams). However, this does not handicap the $\frac{1}{1\oplus D\oplus D^2}$ precoder since only the weight-three event of length three corresponds to the squared distance $d^2_E = 2$, and this event is easy to prevent with the interleaver. Thus, we can expect that $d^2_{E,\min} \geq 4$ for codes employing this precoder. Weight-four error events can be considered to be two weight-two events, and higher-weight events either have large Euclidean distances or very small multiplicities. Thus, we need not concern ourselves with error events of weight four or higher for any of the precoders.

Lastly, we remark that, while the $\frac{1}{1\oplus D}\cdot\frac{1}{1\oplus D^2}$ precoder enjoys the best conditional distance spectrum, it increases the number of states to eight for the APP detector, as compared to four for the other three precoders.

3. Performance of the New Codes

Fig. 5 presents simulation results for rate 16/17 and 64/65 SCC's for the new precoders, where the $\frac{1}{1\oplus D^2}$ curves of Fig. 2 have been included for comparison. We observe that all three new precoders offer vastly superior performance for the rate 16/17 case and that there is no hint of a floor below $P_b = 10^{-7}$ for the $\frac{1}{1\oplus D\oplus D^2}$ and $\frac{1}{1\oplus D}\cdot\frac{1}{1\oplus D^2}$ precoders. For rate 64/65 these precoders are still superior to the $\frac{1}{1\oplus D^2}$ precoder, and the floor for the $\frac{1}{1\oplus D}\cdot\frac{1}{1\oplus D^2}$ precoder appears nowhere near $P_b = 10^{-7}$.

As further demonstration of the efficacy of the $\frac{1}{1\oplus D\oplus D^2}$ and $\frac{1}{1\oplus D}\cdot\frac{1}{1\oplus D^2}$ precoders, we have plotted in Fig. 6 the performance of the rate 16/17 SCC of Fig. 2 against that of SCC's which employ these precoders together with the memory-two (7,5) RSC code as an outer code (punctured to rate 16/17). We observe that both memory-two codes are actually superior to the memory-four code of Fig. 2. The code which utilizes the $\frac{1}{1\oplus D\oplus D^2}$ precoder is particularly attractive since the inner code (precoded PR4 channel) and the outer code (memory-two RSC code) require only four-state APP processors.

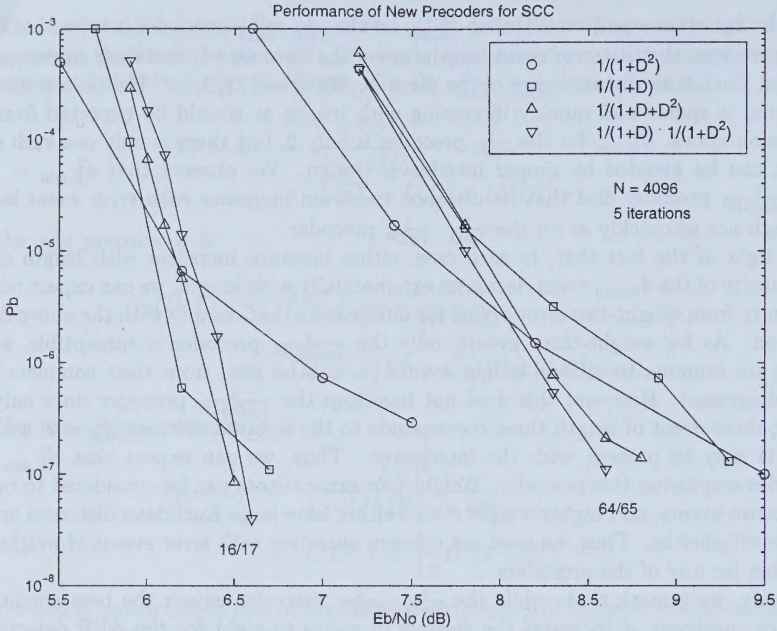


Fig. 5. Performance comparison of the new precoders for rate 16/17 and 64/65 SCC's.

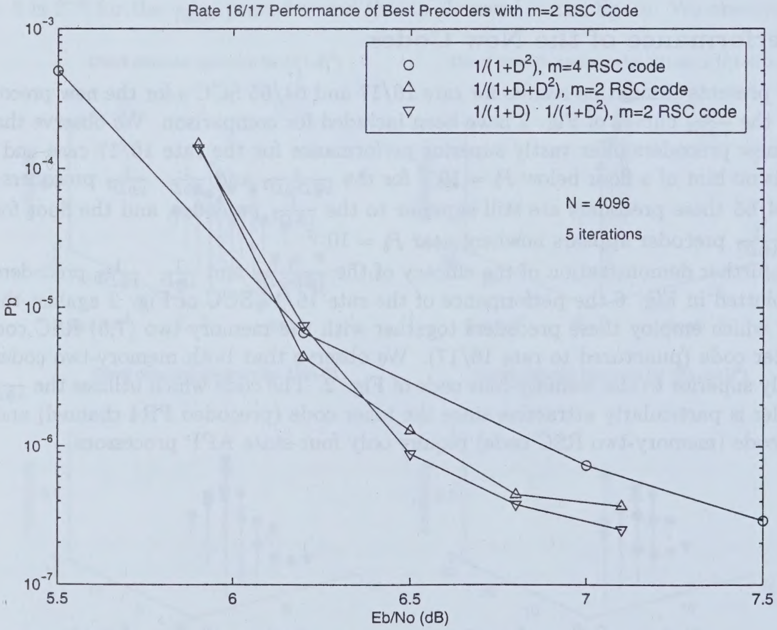


Fig. 6. Performance comparison of the best two precoders employing a memory-two, rate 16/17 outer RSC code and the memory-four rate 16/17 SCC code of Fig. 2.

References

[1] C. Berrou, A. Glavieux, and P. Thitimajshima, “Near Shannon limit error- correcting coding and decoding: Turbo codes,” *Proc. 1993 Int. Conf. Comm.*, pp. 1064- 1070.

[2] W. Ryan, “Performance of high rate turbo codes on a PR4-equalized magnetic recording channel,” *Proc. 1998 Int. Conf. on Comm.*

[3] W. Ryan, L. McPheters, and S. McLaughlin, “Combined turbo coding and turbo equalization for PR4-equalized Lorentzian channels,” *1998 Conf. on Information Systems and Sciences.*

[4] W. Ryan, “Optimal code rates for concatenated codes on a PR4-equalized magnetic recording channel,” *Proc. Conf. on Inf. Sciences and Systems*, pp. 432-437, Mar. 1999.

[5] L. McPheters, S. McLaughlin, and K. Narayanan, “Precoded PRML, serial concatenation, and iterative (turbo) decoding for magnetic recording,” *1999 Int. Conf. on Magnetics.*

[6] L. McPheters, S. McLaughlin, and E. Hirsch, “Turbo codes for PR4 and EPR4 magnetic recording,” *Proc. of the 1998 Asilomar Conf. on Computers and Communications,*” Pacific Grove, CA, pp. 1778-1782, Nov. 1998.

[7] T. Souvignier, A. Friedmann, M. Öberg, P. Siegel, R. Swanson, and J. Wolf, “Turbo decoding for PR4: parallel versus serial concatenation,” *Proc. 1999 Int. Conf. on Comm.*

[8] M. Öberg and P. Siegel, “Performance analysis of turbo-equalized dicode partial-response channels,” *Proc. 36th Allerton Conf. on Communication, Control, and Computing* (Sept. 1998).

[9] D. Raphaeli and Y. Zarai, “Combined turbo equalization and turbo decoding,” *Proc. Int. Symp. on Turbo Codes*, pp. 180-187, Sept. 1997.

[10] L. Bahl, J. Cocke, F. Jelinek, and J. Raviv, “Optimal decoding of linear codes for minimizing symbol error rate,” *IEEE Trans. Inf. Theory*, pp. 284-287, Mar. 1974.

[11] J. Wolf and G. Ungerboeck, “Trellis coding for partial response channels,” *IEEE Trans. Comm.*, pp. 765-773, Aug. 1986.

[12] D. Divsalar and F. Pollara, “Multiple turbo codes for deep-space communications,” JPL TDA Progress Report, 42-121, May 15, 1995.

[13] K. Narayanan and G. Stuber, “A serial concatenation approach to iterative demodulation and decoding,” *IEEE Trans. Comm.*, pp. 956-961, July 1999.

[14] P. Hoeher and J. Lodge, “Turbo DPSK: Iterative differential PSK demodulation and channel decoding,” *IEEE Trans. Comm.*, pp. 837-843, June 1999.

[15] M. Peleg and S. Shamai, “Iterative decoding of coded and interleaved noncoherent multiple symbol detected DPSK,” *Electronic Letters*, pp. 1018-1020, June 1997.

[16] E. Zehavi and J. Wolf, “On the performance evaluation of trellis codes,” *IEEE Trans. Inf. Theory*, pp. 196-201, Mar. 1987.

Turbo Codes for Two-track Magnetic Recording Systems

E. Kurtas[*] and T.M. Duman[†]

Abstract

The current research on the application of turbo coded schemes to magnetic recording channels have concentrated on single track systems. As the demand for high data rate systems becomes more pressing, the research on alternative technologies such as multi-track systems become more relevant. In this work, we consider a two-track two-head system combined with turbo encoding and iterative decoding. We investigate the effect of inter-track interference (ITI) on various turbo coded and/or equalized systems. The goal of this work is to point an unexplored feature of turbo codes, namely their ability to combat both deterministic and stochastic ITI as well as providing high gains.

1 Introduction

In single-track magnetic recording systems data is coded, modulated and written on tracks independently from neighboring or any other tracks, with the same modulation code used on each track. The main idea of a multi-track system is to encode, write and read in parallel. The multi-track system under investigation is modeled as a set of parallel partial-response (PR) channels. Each channel is of the form $G^2(1-D)(1+D)^n$. In addition to this, the effect of the interference from each track to the others is characterized by inter-track interference (ITI) coefficients [1, 2].

The goal of any coding scheme employed in recording systems is to increase the areal density (density in bits per unit area). Increase in areal density in the radial direction can be achieved by narrowing the track width. Some of the drawbacks of such an approach are the loss in SNR, increased ITI and diffuculty of building robust servo systems. The effect of increased ITI can be reduced by means of multiple recording heads [3, 4] and as shown in [1, 2] by designing specific codes which eliminate the sequences that result in performance degradation.

---

[*] Quantum Corp., Shrewsbury, MA 01545 USA Tel : (508) 770-3822, Fax: (508) 770-2604 e-mail: ekurtas@tdh.qntm.com

[†] Telecommunications Research Center, Electrical Engineering Department, Arizona State University, Tempe, AZ, 85287-7206 USA

After the introduction of turbo coding-decoding schemes [5, 6], various researchers [7, 8, 9, 10, 11, 12, 13, 14] have applied the turbo methods to magnetic recording channels with great success. However, all the efforts have concantrated on single-track systems. The goal of this work is to design and analyze the turbo coded multi-track systems with special emphasis given to two-track magnetic recording channels.
Section II describes the multi-track system under investigation and the various turbo coding-decoding methods considered in this work. Section III provides detailed simulation results for a two-track two-head system for both deterministic and stochastic ITI cases. Finally, section IV concludes the paper.

2 System Model

In this section we first describe the multi-track recording channel under investigation and then explain the encoding and decoding schemes involved. Throughout the work we assume, without loss of generality, the number of tracks and the heads flying over them to be equal to M and the all the tracks to be bit-synchronous.

If we let $\mathbf{a}_n = (a_n^1, a_n^2, ..., a_n^M) \in \{\pm 1\}^M$ correspond to the information vector written on the M tracks at time nT with T given as the channel bit duration, $h^{im}(t)$ to the impulse response of the channel between i-th track and m-th receiver head, $\alpha^{im}(t)$ to the ITI coefficients representing the effect of the m-th track on the i-th track and finally $\eta^i(t)$ to the AWGN term for the i-th track with distribution $\mathcal{N}(0, \sigma_i^2)$, then the read-back signal (read by the i-th head) corresponding to i-th track for $\forall i \in \{1, 2, \ldots, M\}$ can be expressed as,

$$\begin{aligned} r^i(t) &= \sum_{m=1}^{M} \alpha^{im}(t) \sum_{n=-\infty}^{\infty} a_n^m h^{im}(t - nT) \\ &+ \eta^i(t) \end{aligned} \qquad (1)$$

It can be shown [15, 16] that a set of sufficient statistics for maximum-likelihood detection can be obtained by passing each received signal through an appropriate matched-filter followed by sampling at T intervals. If the sampled outputs are further filtered for noise whitening [16], the resulting equivalent discrete-time channel model becomes,

$$y_k^i = \sum_{m=1}^{M} \alpha_k^{im} \sum_{n=0}^{L^{im}-1} a_{k-n}^m f_n^{im} + \eta_k^i \qquad (2)$$

where we assumed $f^{ij} = 0$ when $n < 0$ or $n > L^{ij}, \forall i, j \in \{1, 2, \ldots, M\}$, i.e. the channel spans are finite. We use this discrete-time eqivalent model throughout the work presented here.

The overall multi-track recording system structure is shown in Figure 1. The bits for each track are encoded by a turbo encoder and then passed through a channel permuter followed by an optional precoding operation. Figure 2 shows the decoder structure when there is no feedback from turbo encoders to the multi-track decoder ("Partial" Turbo [12]). The scheme where there is feedback (i.e., Turbo Equalization) between indivudual turbo decoders and the multi-track decoder is shown in Figure 3 ("Full" Turbo [12]). Finally, the last system we consider uses convolutional encoders (instead of turbo) for each track and the only iterations are between soft-convolutional decoders of each track and the multi-track decoder ("Serial" Turbo [12]) as shown in Figure 4.

3 Simulation Results

For the simulations, we have considered a two-track two-head recording system. Each track-head combination has been modeled as an ISI channel with discrete time equivalent impulse response given as $F^{ij}(D) = \frac{(1-D^2)}{\sqrt{(2)}}$, $\forall i,j$.We have run extensive simulations to analyze the performance of iterative multi-track multi-head schemes proposed in this work under various ITI conditions. We used (23/31) recursive systematic convolutional component codes for the turbo encoders and after puncturing the overall rate was $\frac{4}{5}$. The block length for turbo permuters was set as $N = 5120$. All the systems considered here used a $1/(1 \bigoplus D^2)$ type precoder. The SNR per track for two-track case with symmetric ITI,$\alpha_k^{ij} = \alpha$, $\forall i,j,k$ can be defined as,

$$SNR = 10log(\frac{(1+\alpha^2)||F(D)||^2}{R * N0}) \tag{3}$$

when ITI is deterministic, and as,

$$SNR = 10log(\frac{(1+\sigma_{\alpha}^2)||F(D)||^2}{R * N0}) \tag{4}$$

when ITI is modeled as a stationary stochastic process. Figure 5, 6, and 7 presents the results for partial, full and serial turbo systems for deterministic and symmetric ITI with $\alpha^{12} = \alpha^{21} = \alpha = 0.0$, $\alpha = 0.25$ and $\alpha = 0.5$, respectively. We make the following observations with respect to coding gains of these turbo systems,

- The serial system performs about 0.2dB better than the full turbo system, which in turn comes ahead of partial turbo system with approximately same amount for all of the deterministic ITI cases ($\alpha^{12} = \alpha^{21} = \alpha = 0.0$, $\alpha = 0.25$ and $\alpha = 0.5$).
- The net coding gain provided by the serial system improves from around 4.5dB to approximately 6dB as α goes from 0 to 0.5 for operating BER of 10^{-4}.

- The coding gains provided by the two-track codes of [2] ranges from 3dB to 4.8dB as α goes from 0 to 0.5 without calculating for the rate loss. For a rate $\frac{4}{5}$ code the net gains of these codes become 2dB to 3.8dB approximately. Therefore the turbo coded systems studied in this work surpass the codes of [2] by 2.5dB to 2.2dB (net) as α goes from 0 to 0.5 for operating BER of 10^{-4}.
- The comparisons for lower than 10^{-4} operating BER points show that the gains of turbo coded systems increase over uncoded systems.

When the inter-track interference is rapidly varying or there is no way to reliably estimate ITI coefficients, multi-track detectors can not make any judgment on the value of ITI coefficients. Thus the key issue becomes the robustness of the multi-track coding/decoding schemes introduced here. For this purpose, we investigated the robustness of the multi-track iterative decoders introduced here under rapidly changing inter-track interference conditions. We have introduced stochastic ITI with Gaussian distribution N(0, α^2) for both turbo coded and uncoded systems. Due to the Gaussian nature of the ITI the multi-track detector assumed no prior knowledge of ITI values, and assumed $\alpha^{12} = \alpha^{21} = \alpha = 0$. Although we could try to estimate these values in turbo coded schemes, our simulations show even without any estimation turbo schemes perform almost perfectly (i.e as if the actual ITI values were known). This is quite to the opposite of the performance exhibited by uncoded multi-track scheme. The simulation results presented in Figure 8, 9 and 10 suggest that,

- The uncoded multi-track detector considers the stochastic ITI as pure noise and saturates as the ITI variance increases.
- It can be seen from Figure 8, 9 and 10 that all of the turbo systems deal very well with the unknown ITI and able to eliminate the degradation totally.
- The relative performance of serial, full and partial turbo systems are similar to the deterministic ITI results.

4 Conclusion

In this work we have designed and analyzed the performance of multi-track iterative decoders for magnetic recording channels. We have first shown when ITI coefficients are estimated perfectly iterative decoders provide 4.5dB to 6dB over uncoded system which uses an optimal multi-track decoder. Then we have provided results for the stochastic ITI case to investigate the robustness of turbo coded/equalized schemes. Our results indicate that even under severe ITI conditions where conventional uncoded multi-track decoders saturate, the iterative decoders proposed in this work show very robust and superior performance.

References

[1] Erozan Kurtas, John Proakis and Masoud Salehi, ' Coding for Multi-track Magnetic Recording Systems', *IEEE Transactions on Information Theory*, vol.43, no.6, pp.2020-2023, November 1997.

[2] E. Soljanin and C. N. Georghiades, ' Coding for Two-head Recording Systems', *IEEE Transactions on Information Theory*, May 1995,pp.747-755.

[3] L. C. Barbosa, ' Simultaneous Detection of Readback Signals from Interfering Magnetic Recording Tracks Using Array Heads', *IEEE Trans. Magn.*, vol. MAG-26, no. 5, pp. 2163-2165, Sept. 1990.

[4] R. E. Swanson and J. K. Wolf, ' A New Class of Two-dimensional RLL Recording Codes', *IEEE Trans. Magn.*, vol. MAG-28, no. 6, pp. 3407-3416, Nov. 1992.

[5] Berrou C. and Glavieux A., "Near Optimum Error Correcting Coding and Decoding: Turbo-Codes", *IEEE Transactions on Communications*, vol. 44, no. 10, pp. 1261-1271,October, 1996

[6] Hagenauer J., "Iterative Decoding of Binary Block and Convolutional Codes", *IEEE Transactions on Information Theory*,vol. 42, no. 2, pp. 429-445,March,1996

[7] C. Heegard, "Turbo Coding in Magnetic Recording", *Proceedings of the Information Theory Workshop*, San Diego, CA, 1998.

[8] W. E. Ryan, L. L. McPheters, and S. McLaughlin, "Combined Turbo Coding and Turbo Equalization for PR4Equalized Lorentzian Channels", *Proceedings of the Conference on Information Sciences and Systems*, Mar. 1998.

[9] W. Pusch, D. Weinrichter, and M. Tafarner, "Turbo Codes Matched to the $1 - D^2$ Partial Response Channel", *Proceedings of the International Symposium on Information Theory*, 1998, p. 62.

[10] M. C. Reed and C. B. Schelegel, "An Iterative Receiver for the Partial Response Channel", *Proceedings of International Symposium on Information Theory*, 1998, p. 63.

[11] W. E. Ryan, "Performance of High Rate Turbo Codes on a PR4 Equalized Magnetic Recording Channel", *Proceedings of IEEE International Conference on Communications*, June 1998.

[12] T. Souvignier, A. Friedman, M. Oberg, P. H. Siegel, R. E. Swanson, and K. K. Wolf, "Turbo Codes for PR4: Parallel versus Serial Concatenation", accepted to *1999 IEEE International Conference on Communications*.

[13] T. Duman and E. Kurtas, "Performance of Turbo Codes over Magnetic Recording Channels ", accepted to MILCOM 1999.

[14] T. Duman and E. Kurtas,'Comprehensive Performance Investigation of Turbo Codes over High Density Magnetic Recording Channels', accepted to GLOBECOM 1999.

[15] S. Verdú, " Minimum Probability of Error for Asynchronous Gaussian Multiple-Access Channels", *IEEE Trans. Inform. Theory,* vol. IT-32, pp. 85-96, january 1986.

[16] J.G. Proakis, *Digital Communications,* New York, NY:McGraw-Hill, 1983.

[17] Robertson P. and Villebrun E. and Hoeher P.,"A Comparison of Optimal and Sub-Optimal MAP Decoding Algorithms Operating in the Log Domain", *Proceedings ofIEEE Int. Conf. on Comm. (ICC)*,pp. 1009-1013,1995

[18] Robertson P., " Illuminating the Structure of Code and Decoder of Parallel Concatenated Recursive Systematic (Turbo) Codes", *Proceedings of IEEE GLOBECOM Conference,* pp. 1298-1303, 1994

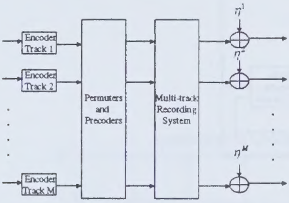

Encoder Track 1
Encoder Track 2
Encoder Track M
Permuters and Precoders
Multi-track Recording System
η^1
η^2
η^M

Figure 1: Multi-track Recording System

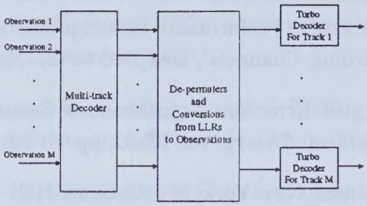

Observation 1
Observation 2
Observation M
Multi-track Decoder
De-permuters and Conversions from LLRs to Observations
Turbo Decoder For Track 1
Turbo Decoder For Track M

Figure 2: Iterative multi-track decoder without turbo equalization (Partial Turbo:PRT)

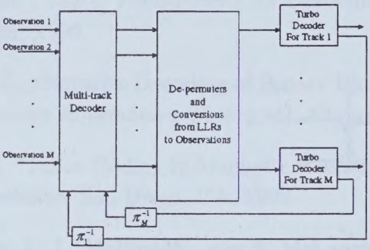

Observation 1
Observation 2
Observation M
Multi-track Decoder
De-permuters and Conversions from LLRs to Observations
Turbo Decoder For Track 1
Turbo Decoder For Track M
π_M^{-1}
π_1^{-1}

Figure 3: Iterative multi-track decoder with turbo equalization (Full Turbo: FT)

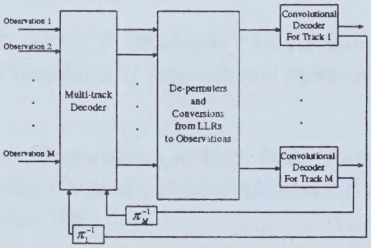

Observation 1
Observation 2
Observation M
Multi-track Decoder
De-permuters and Conversions from LLRs to Observations
Convolutional Decoder For Track 1
Convolutional Decoder For Track M
π_M^{-1}
π_1^{-1}

Figure 4: Convolutinal coded and turbo equalized multi-track system (Serial Turbo:SRT)

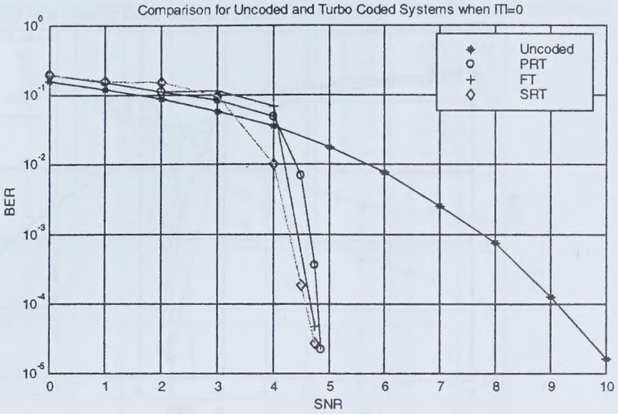


Figure 5: Comparisons for rate=4/5 partial, full and serial turbo coded systems with uncoded for $\alpha = 0.0$

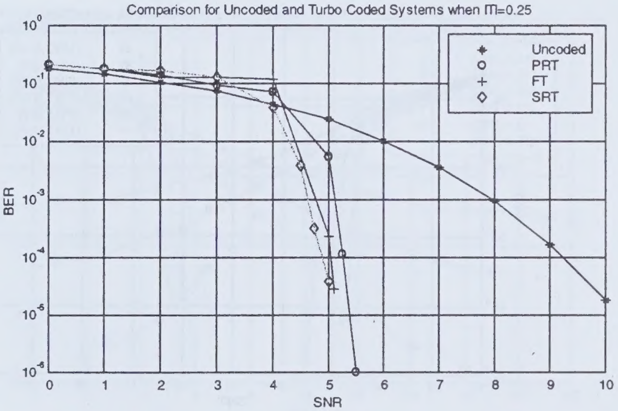


Figure 6: Comparisons for rate=4/5 partial, full and serial turbo coded systems with uncoded for $\alpha = 0.25$

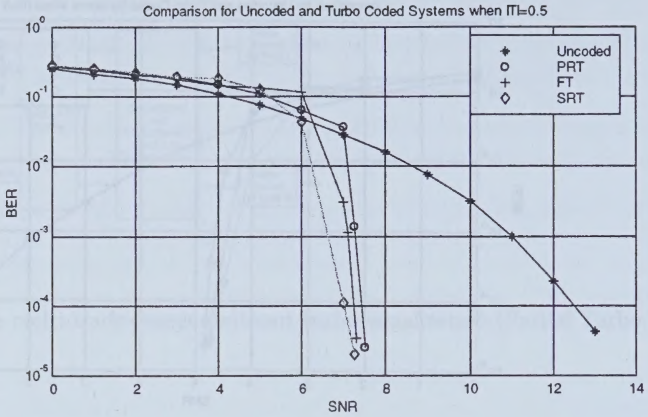


Figure 7: Comparisons for rate=4/5 partial, full and serial turbo coded systems with uncoded for $\alpha = 0.5$

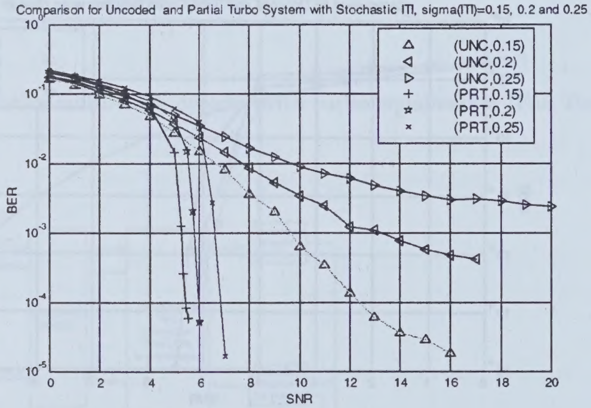


Figure 8: Partial Turbo system with Normal distributed ITI (N(0, α^2)) for $\alpha = 0.15, 0.2,$ and 0.25

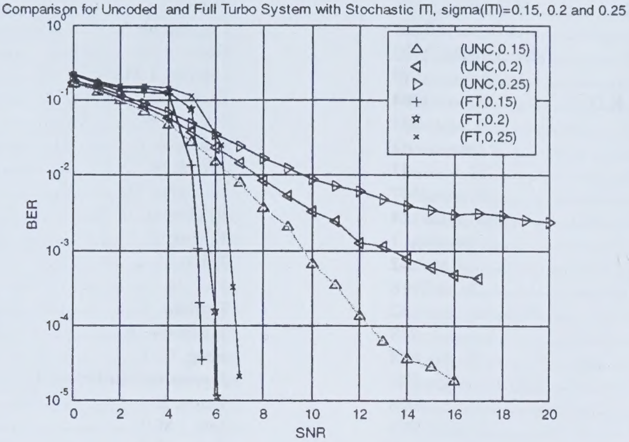


Figure 9: Full Turbo system with Normal distributed ITI (N(0, α^2)) for $\alpha = 0.15, 0.2,$ and 0.25

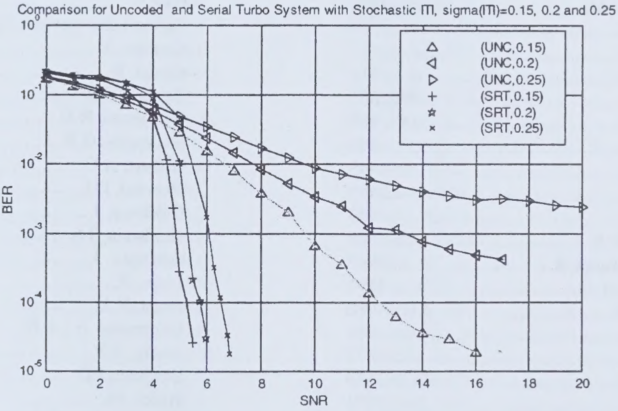


Figure 10: Serial Turbo system with Normal distributed ITI (N(0, α^2)) for $\alpha = 0.15, 0.2,$ and 0.25

LIST OF AUTHORS

Abdelwahed, S. 85
Ailawadhi, V. 1201
Altman, E. 1085, 1132
Andonovic, I. 679
Anim-Appiah, K.D. 1304
Aplevich, J.D. 925
Armony, M. 42
Ashikhmin, A. 52
Atar, R. 607
Ayyorgun, S. 124
Baccelli, F. 1
Bambos, N. 42
Barbarossa, S. 1016
Barg, A. 52
Baron, D. 728
Baruch, A. 328
Baum, C.W. 780
Baum, K.L. 1026
Bengtsson, M. 987
Bhattacharjee, S. 147
Bhulai, S. 1132
Bialek, W. 368
Biglieri, E. 851
Bilar, D. 348
Birk, Y. 728
Block, F.J. 780
Blumenthal, D.J. 414, 689
Boixadera, F. 216
Boroson, D.M. 1221
Borst, S. 316
Boutros, J. 515
Boxma, O. 316
Caire, G. 790, 851
Calvert, K. 147, 166
Carlemalm, C. 800
Carroll, C. 443
Chae, Y. 147
Chan, A. 443
Chan, V.W.S. 388
Chandrasekaran, R. 967
Chang, J.-H. 1191
Chang, W. 280
Chen, B. 881
Chen, J. 471
Chia, M.C. 679
Cho, J.H. 871
Choi, S. 157
Chong, E.K.P. 485, 495
Chu, M.J. 525
Collins, B.E. 709
Cruz, R.L. 124, 709
Cybenko, G. 348, 748
Darabi, H. 1036
Das, D. 1211
Decasper, D. 157, 768
Dehart, J. 157
Denker, G. 738
Dogandžić, A. 948
Down, D.G. 280
Duman, T.M. 1334
Dumer, I. 61
Dupuis, P. 607
Duursma, I. 555
Effros, M. 378
El Gamal, H. 186, 445
El Ghaoui, L. 655
Ellinas, G. 659
Evans, J. 758
Fan, J.L. 70, 1314
Farmine, Y. 216
Feldmann, A. 30
Feng, G.-L. 545
Fernández-Gaucherand, E. 627
Feron, E. 655
Fitz , M.P. 897
Foley, R.D. 280
Fossorier, M.P.C. 461, 471
Frey, B.J. 241
Friedmann, A. 1314
Fuks, G. 977
Gaborit, P. 535
Gajic, Z. 475
Ganesh, A.J. 288
Gao, J. 1201
Gao, Y. 935
Garcia-Luna-Aceves, J.J. 738
Gaubert, S. 1
Gaujal, B. 1132
Gelfand, S.B. 915
Georganas, N.D. 1113
Giannakis, G.B. 770, 1016
Gilbert, A.C. 30
Goeckel, D.L. 525
Goldberg, J. 977
Goldstein, J.S. 196
Griffioen, J. 166
Guan, X. 1042
Guan, Y.L. 451
Hammons, Jr., A.R. 186, 445
Hanly, S.V. 1181
Herdtner, J.D. 485
Hicks, M. 127
Hjálmtýsson, G. 137
Ho, P. 1152
Hochwald, B.M. 1266, 1284
Holloway, L.E. 1042
Hong, D. 1
Honig, M.L. 810, 1236
Hordijk, A. 1132
Hottinen, A. 204
Huffman, W.C. 535
Hunter, D.K. 679

Jafari, M.A. 1036
Jelenković, P. 316
Ji, T. 1171
Jiang, G. 348
Jiang, S. 103
Jiménez, T. 1085
Jin, H. 893
Joo, Y. 30
Kawamura, R. 176
Ke, N.-P. 641
Keller, R. 157
Keskinoz, M. 1294
Khandekar, A. 891
Kim, J.H. 1105
Kim, J.-L. 535
Kocaoglan, E. 937
Konstantopoulos, T. 21
Koole, G. 1085
Kotopoulos, C. 567
Krauss, T.P. 1274
Krogh, B.H. 905
Krogmeier, J.V. 915
Kulkarni, A. 758
Kumar, B.V.K.V. 1294
Kumar, R. 103
Kumar, S. 597
Kurtas, E. 1314, 1334
Kurtz, T.G. 11
Lang, J.P. 689
Last, G. 21
Le Pézennec, Y. 216
Learned, R.E. 1221
Lee, E.B. 637, 651
Lehnert, J.S. 871
Li, G. 451
Liberty, S.R. 617
Likhanov, N. 567
Lin, S. 461, 471
Lin, T. 400
Litsyn, S. 52
Liu, Y. 461, 897
Liu, Z. 1016
Lockwood, J. 157
Logothetis, A. 800, 1256
Lu, B. 214
Lu, Y.L. 997
Luo, J.C. 637
Lyshevski, S.E. 937
MacKay, D.J.C. 241
Maglaras, C. 587
Maharaj, H. 903
Malisoff, M. 657
Mandayam, N. 475
Mandelbaum, A. 270, 1095
Mandjes, M. 1105
Mantravadi, A. 424
Marbach, P. 1075
Marcus, B. 70
Marzetta, T.L. 958, 1284
Massey, W.A. 270, 1095
Mazumdar, R.R. 567
McEliece, R.J. 891, 893
McLaughlin, S.W. 901, 1304, 1314
Merched, R. 338
Merhav, N. 328
Merugu, S. 147
Meseguer, J. 738
Messer, H. 977
Minden, G. 758
Minhas, R.S. 75
Mirkovic, J. 113
Mitra, U. 790
Mitter, S. 940
Mitzenmacher, M. 326
Modiano, E. 390
Moore, J.T. 127
Mortazavian, H. 113
Müller, R.R. 1246
Mullins, B. 166
Myrick, W.L. 196
Narayanan, K.R. 899
Narula-Tam, A. 390
Nehorai, A. 948
Nettles, S. 127
Ng, B.P. 997
O'Connell, N. 288
O'Sullivan, M.E. 557
Okino, C. 748
Olejniczak, K.J. 935
Ölveczky, P.C. 738
Ottersten, B. 987
Pan, R. 1122
Papadias, C.B. 1284
Paschalidis, I.Ch. 306
Pereira, F.C. 368
Perros, H.G. 669
Pfister, H.D. 260
Pham, K.D. 617
Pidaparti, B.R. 937
Pless, V. 535
Poor, H.V. 800, 1256
Popescu, D.C. 505
Pottie, G. 1201
Pouttu, A. 434
Prabhakar, B. 288, 1122
Pursley, M.B. 1162
Rajan, D. 204
Raju, J. 738
Ramanan, K. 297
Ratasuk, R. 1236
Reiman, M.I. 270, 1095
Ren, Z. 905
Ribeiro, V. 30
Richardson, T. 221, 231
Rose, C. 505
Roth, R. 70

Rouskas, G.N. 669
Ryan, W.E. 1324
Sain, M.K. 617
Sanders, M. 147
Sarkar, S. 718
Sasaki, G. 400
Savari, S.A. 358
Sayed, A.H. 338
Sayeed, A.M. 1006
Scaglione, A. 1016
See, C.M.S. 997
Sehgal, A. 166
Sen, A. 402
Shah, T. 402
Shahrrava, B. 925
Shamai, S. 841
Shayman, M.A. 627
Sheth, S. 758
Shroff, N.B. 495
Shynk, J.J. 967
Siegel, P.H. 260
Siler, M. 1142
Sin, Y.K. 997
Sinha, A.S.C. 937
Sinha, B.P. 402
Sivasankaran, R. 901
Smith, B. 738
Sohrabi, K. 1201
Song, L. 475
Spencer, Jr., B.F. 617
Stadler, R. 176
Stark, W.E. 525, 1171
Stockbridge, R.H. 11
Stolyar, A.L. 297, 1095
Sweldens, W. 1266
Talcott, C.L. 738
Tang, H. 461
Tanner, R.M. 249
Tantinarawat, K. 1006
Taricco, G. 851
Tassiulas, L. 718, 1191
Tatikonda, S. 940
Tepedelenlioğlu, C. 770
Thomas, T.A. 1026
Tishby, N. 368
Tong, L. 881
Tonguz, O.K. 699
Tse, D.N.C. 861, 1055, 1152, 1181
Tsybakov, B. 1113
Turner, J. 157
Urbanke, R. 221, 231
Ushio, T. 93
Vadde, V. 1294
Varanasi, M.K. 820, 1211
Varvarigos, E.A. 689
Vassilaras, S. 306
Veeravalli, V.V. 424, 831
Verdú, S. 841
Vialle, S. 515
Vijayakumar, R. 1065
Viterbo, E. 851
Vöcking, B. 326
Vook, F.W. 1026
Walrand, J. 1142, 1152
Wang, X. 214
Wasserman, K.M. 1065
Wei, Y. 915
Weiss, G. 577
Wen, S. 166
Whinnett, N. 216
Wichman, R. 204
Wilkins, C.S. 1162
Willinger, W. 30
Willsky, A.S. 1221
Wolf, T. 157, 768
Wonham, W.M. 75, 85
Wood, J.A. 895
Woodward, G. 1236
Xiao, M. 495
Xiao, W. 810
Xu, C. 471
Yamamoto, S. 93
Yaz, E.E. 935
Yin, J. 997
You, K.H. 651
Zaidel, B.M. 841
Zegura, E. 147
Zhang, J. 1006
Zhang, M. 443
Zheng, L. 861
Zhu, Y. 669
Zoltowski, M.D. 196, 1274

UNIVERSITY OF ILLINOIS
LIBRARY
AT URBANA-CHAMPAIGN
ENGINEERING

UNIVERSITY OF ILLINOIS-URBANA

3 0112 043448312